

UNCLASSIFIED

AFIT/ENG/TR96-02

Air Force Institute of Technology

Simulation of a Dynamic Aircraft Radar Signature

Joseph J. Sacchini, Captain, USAF
Assistant Professor of Electrical Engineering

3 June 1996

19960719 090

"Approved for public release; distribution unlimited"

DTIC QUALITY INSPECTED 4

REPORT DOCUMENTATION PAGEForm Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE June 1996	3. REPORT TYPE AND DATES COVERED Technical Report	
4. TITLE AND SUBTITLE Simulation of a Dynamic Aircraft Radar Signature			5. FUNDING NUMBERS	
6. AUTHOR(S) Joseph J. Sacchini, Ph.D. Captain, USAF				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Air Force Institute of Technology, WPAFB OH 45433-6583			8. PERFORMING ORGANIZATION REPORT NUMBER AFIT/ENG/TR96-02	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Mr. Philip Gibber AFOSR/NI 110 Duncan Ave. Suite B115 Bolling AFB, DC 20332-8080			10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; Distribution Unlimited			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) This report describes a package which simulates the dynamic radar signatures a radar receiver would expect when tracking a moving target such as an aircraft or a missile. The package consists of several MATLAB codes. The basic engagement simulated is a radar, such as seen on a fully active missile seeker, engaging a target such as an aircraft. The static scattering characteristics of the target are required as an input to the package. The package then simulates the radar-target engagement dynamics modeling the target motion and the physical target vibrations which effect the signature the radar receives. The outputs of the package are the scattered electric field, the Doppler shift, and the angular tracking error, or glint, all as seen by the radar. This package is meant to aid a hardware-in-the-loop simulation of a radar-target engagement.				
14. SUBJECT TERMS Radar, SAR, Dynamic Signatures, Parametric Modeling			15. NUMBER OF PAGES 81	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT UL	

GENERAL INSTRUCTIONS FOR COMPLETING SF 298

The Report Documentation Page (RDP) is used in announcing and cataloging reports. It is important that this information be consistent with the rest of the report, particularly the cover and title page. Instructions for filling in each block of the form follow. It is important to *stay within the lines* to meet *optical scanning requirements*.

Block 1. Agency Use Only (Leave blank).

Block 2. Report Date. Full publication date including day, month, and year, if available (e.g. 1 Jan 88). Must cite at least the year.

Block 3. Type of Report and Dates Covered. State whether report is interim, final, etc. If applicable, enter inclusive report dates (e.g. 10 Jun 87 - 30 Jun 88).

Block 4. Title and Subtitle. A title is taken from the part of the report that provides the most meaningful and complete information. When a report is prepared in more than one volume, repeat the primary title, add volume number, and include subtitle for the specific volume. On classified documents enter the title classification in parentheses.

Block 5. Funding Numbers. To include contract and grant numbers; may include program element number(s), project number(s), task number(s), and work unit number(s). Use the following labels:

C - Contract	PR - Project
G - Grant	TA - Task
PE - Program Element	WU - Work Unit Accession No.

Block 6. Author(s). Name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. If editor or compiler, this should follow the name(s).

Block 7. Performing Organization Name(s) and Address(es). Self-explanatory.

Block 8. Performing Organization Report Number. Enter the unique alphanumeric report number(s) assigned by the organization performing the report.

Block 9. Sponsoring/Monitoring Agency Name(s) and Address(es). Self-explanatory.

Block 10. Sponsoring/Monitoring Agency Report Number. (If known)

Block 11. Supplementary Notes. Enter information not included elsewhere such as: Prepared in cooperation with...; Trans. of...; To be published in.... When a report is revised, include a statement whether the new report supersedes or supplements the older report.

Block 12a. Distribution/Availability Statement. Denotes public availability or limitations. Cite any availability to the public. Enter additional limitations or special markings in all capitals (e.g. NOFORN, REL, ITAR).

DOD - See DoDD 5230.24, "Distribution Statements on Technical Documents."

DOE - See authorities.

NASA - See Handbook NHB 2200.2.

NTIS - Leave blank.

Block 12b. Distribution Code.

DOD - Leave blank.

DOE - Enter DOE distribution categories from the Standard Distribution for Unclassified Scientific and Technical Reports.

NASA - Leave blank.

NTIS - Leave blank.

Block 13. Abstract. Include a brief (*Maximum 200 words*) factual summary of the most significant information contained in the report.

Block 14. Subject Terms. Keywords or phrases identifying major subjects in the report.

Block 15. Number of Pages. Enter the total number of pages.

Block 16. Price Code. Enter appropriate price code (*NTIS only*).

Blocks 17. - 19. Security Classifications. Self-explanatory. Enter U.S. Security Classification in accordance with U.S. Security Regulations (i.e., UNCLASSIFIED). If form contains classified information, stamp classification on the top and bottom of the page.

Block 20. Limitation of Abstract. This block must be completed to assign a limitation to the abstract. Enter either UL (unlimited) or SAR (same as report). An entry in this block is necessary if the abstract is to be limited. If blank, the abstract is assumed to be unlimited.

AFIT TECHNICAL REPORT AFIT/ENG/TR96-02
Simulation of a Dynamic Aircraft Radar Signature

Joseph J. Sacchini, Captain, USAF, Ph.D.
Assistant Professor of Electrical Engineering
Department of Electrical and Computer Engineering
Air Force Institute of Technology
Wright-Patterson Air Force Base, Ohio 45433 USA
Phone: (513) 255-3636 ext. 4715
e-mail: jsacchin@afit.af.mil

Abstract

This report describes a package which simulates the dynamic radar signatures a radar receiver would expect when tracking a moving target such as an aircraft or a missile. The package consists of several MATLAB codes. The basic engagement simulated is a radar, such as seen on a fully active missile seeker, engaging a target such as an aircraft. The static scattering characteristics of the target are required as an input to the package. The package then simulates the radar-target engagement dynamics modeling the target motion and the physical target vibrations which effect the signature the radar receives. The outputs of the package are the scattered electric field, the Doppler shift, and the angular tracking error, or glint, all as seen by the radar. This package is meant to aid a hardware-in-the-loop simulation of a radar-target engagement.

Contents

1	Introduction	1
2	Radar-Target Engagement Geometry, Dynamics, and Target Scat- tering	4
2.1	Basic Flight Path	4
2.2	Aircraft Scattering Model	7
2.3	Damped Exponential Modeling of C-29 Scale Model	12
2.4	Scattered Field Calculations	19
2.5	Aircraft Vibrations and Motion Models	24
2.5.1	Rigid Body Motion	24
2.5.2	Individual Scattering Center Motion	27
2.6	Summary of Dynamics and Target Scattering	29
3	Simulation Outputs	32
3.1	Scattered Field	32
3.2	Doppler Shift	33
3.3	Angular Error	37
4	Simulation Example	38
5	MATLAB Code Descriptions	48
6	Conclusions and Recommendations	49
7	References	52
8	Appendices	54

List of Figures

1.1	Simulation development methodology.	3
2.1	Radar Target Geometry.	5
2.2	Definition of azimuth (AZ) and elevation (EL) angles for the target. .	11
2.3	Sketch and Dimensions of C-29 one-third scale model.	14
2.4	Limits of angular coverage for raw C-29 ISAR data.	15
2.5	ISAR Image for One-Third C-29 Scale Model.	20
2.6	Model Generated ISAR Image for One-Third C-29 Scale Model. . . .	21
2.7	Individual Scattering Center Motion Plot, 2cm motion.	30
2.8	Individual Scattering Center Motion Plot, 20cm motion.	31
4.1	Path of C-29 aircraft for sample simulation.	42
4.2	Azimuth and elevation angles versus time for sample simulation. . . .	43
4.3	Scattered Electric Field and Angular Error versus time for sample simulation.	44
4.4	Blow-up of Scattered Electric Field and Angular Error versus time for a portion of the sample simulation.	45
4.5	Doppler frequency and Received Power versus time for sample simulation. .	46
4.6	Blow-up of Doppler frequency and Received Power versus time for a portion of the sample simulation.	47
8.1	Radar-target geometry for 2-D imaging.	55
8.2	2-D frequency domain data set for full angular coverage.	57
8.3	2-D frequency domain data set for limited angular coverage.	58

List of Tables

2.1	Simulation Variables.	6
2.2	C-29 ISAR Data Specifications, Unscaled.	15
2.3	C29model.m Data Specifications, Scaled	16
4.1	Parameters for sample simulation.	39
5.1	Listing of MATLAB codes.	48

1. Introduction

This report describes a package which simulates the dynamic radar signatures a radar receiver would expect when tracking a moving target such as an aircraft or a missile. By dynamic, it is meant that the radar and/or target are physically moving. The package consists of several MATLAB codes.

This work was performed at Australia's Defence Science and Technology Organisation (DSTO), in Salisbury, South Australia, Australia, under the United States' Air Force Office of Scientific Research Scientist and Engineer Exchange Program, from February 1, 1996 to May 10, 1996, by the author. The office within DSTO where the work was performed is The Maritime and Aeronautical Research Laboratory, Guided Weapons Division, Radio Frequency Seekers Group. The supervisor at DSTO was Dr. Noel Martin, Principal Research Scientist and Head of the Radio Frequency Seekers Group.

The basic engagement simulated is a mono-static radar, such as seen on a fully active missile seeker, engaging a target such as an aircraft. The radar is assumed to be mono-static (co-located receive and transmit antenna). The static (no-motion) scattering characteristics of the target are required as an input to the package. These are available for a representative target, the C-29 transport aircraft. The package uses a simplified model for the static scattering from the target, the 2-D Damped Exponential Model. This model is sometimes called the multi-point scattering model and is described in several references [1, 2, 3].

The package then simulates the radar-target engagement dynamics modeling the target motion and the physical target vibrations which will effect the signatures the missile receives. The outputs of the package are the scattered electric field, the Doppler shift, and the angular tracking error, or glint, all as seen by the radar. This package is meant to aid a hardware-in-the-loop simulation of a radar-target

engagement. Figure 1.1 shows a flow diagram outlining the development of this simulation package.

The notational conventions followed in this report are that vectors, or quantities with a magnitude and direction, are so noted by the “vec” notation, for example, the vector $\vec{R}t$. Unit vectors, or vectors with unit magnitude, are so designated by a “hat”, for example the unit vector $\hat{D}t$. Also, complex quantities are denoted by boldtype, for example, the complex number \mathbf{p}_x . The imaginary basis is denoted by j , i.e. $j = \sqrt{-1}$. Scalar quantities do not have the “vec” sign and are not boldtyped.

Throughout this report, the term radar is used to describe what is tracking the target. The radar can be a conventional radar as well as a missile seeker. Also, the radar can be stationary or on a moving platform such as an aircraft. The term target describes what is being tracked by the radar. The target can be an aircraft, or any flying or stationary target. In this report, the radar is always called the radar while the target is usually called the target, but is also called the aircraft in many places, since the target shown in the simulations is a C-29 transport aircraft.

This report is organized as follows. Section 2 covers the model used for the target scattering and defines the dynamic scenario between radar and target. Section 3 defines the output quantities of the simulations. Section 4 shows the results of a typical simulation. Section 5 describes the MATLAB codes used to implement the simulations. Finally, Section 6 concludes the report and provides recommendations for further research in the modeling of dynamic radar data.

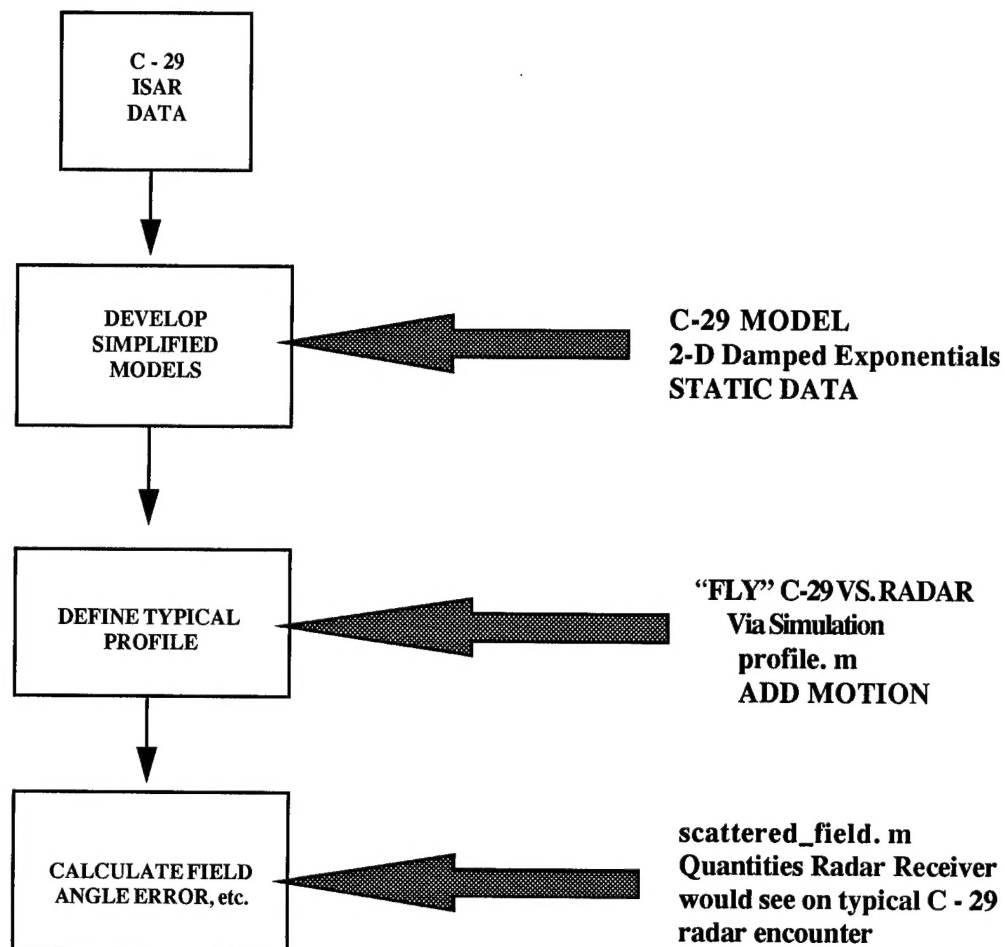


Figure 1.1: Simulation development methodology. First, simplified static scattering models are developed. Then motion compensation is added to the models. Finally, the quantities of interest are calculated.

2. Radar-Target Engagement Geometry, Dynamics, and Target Scattering

This section describes the radar-target dynamics and the scattering model used for the target. Along with the descriptions of the dynamic situations and target scattering calculations, examples are used to demonstrate the theory.

2.1 Basic Flight Path

The basic engagement geometry is shown in Figure 2.1. The variables used in the engagement geometry are described in Table 2.1. Please note that all units are in the metric system, i.e., all angles are in degrees, all frequencies are in hertz, all distances in meters, all velocities in meters per second, all powers in watts, and all gains are in numeric (not in dB).

For the simulations it is possible that both the radar and target are moving. In a real situation, of course, this is the case, but for the simulations performed here it is assumed that the radar is fixed at the origin of the coordinate system. Thus, a radar fixed coordinate system is assumed and $\vec{R}o = (0, 0, 0)$. Also, it is assumed that the radar illuminating the target is continuous-wave (CW), having a 100% duty cycle. If a pulsed radar is desired, the results of the simulations can easily be converted to account for this [4]. The angles AZ and EL are relative to the waterline, or centerline, of the target, defined by the direction vector ($\hat{P}t$).

The simulations are set up so that at a given instant in time, t , the target is at a given position, $\vec{R}t$, traveling in the $\hat{D}t$ direction at velocity Vt . The target is assumed to be pointing in the $\hat{P}t$ direction. By pointing, it is meant that the physical centerline, or waterline, of the target is in the $\hat{P}t$ direction. For now, we assume the $\hat{D}t = \hat{P}t$. In Section 2.3, roll, pitch, and yaw angles are introduced to allow $\hat{D}t$ and $\hat{P}t$ to take on different values. Next, all of the simulation outputs are calculated. Then time is incremented by dt , the target (aircraft) moves the appropriate distance,

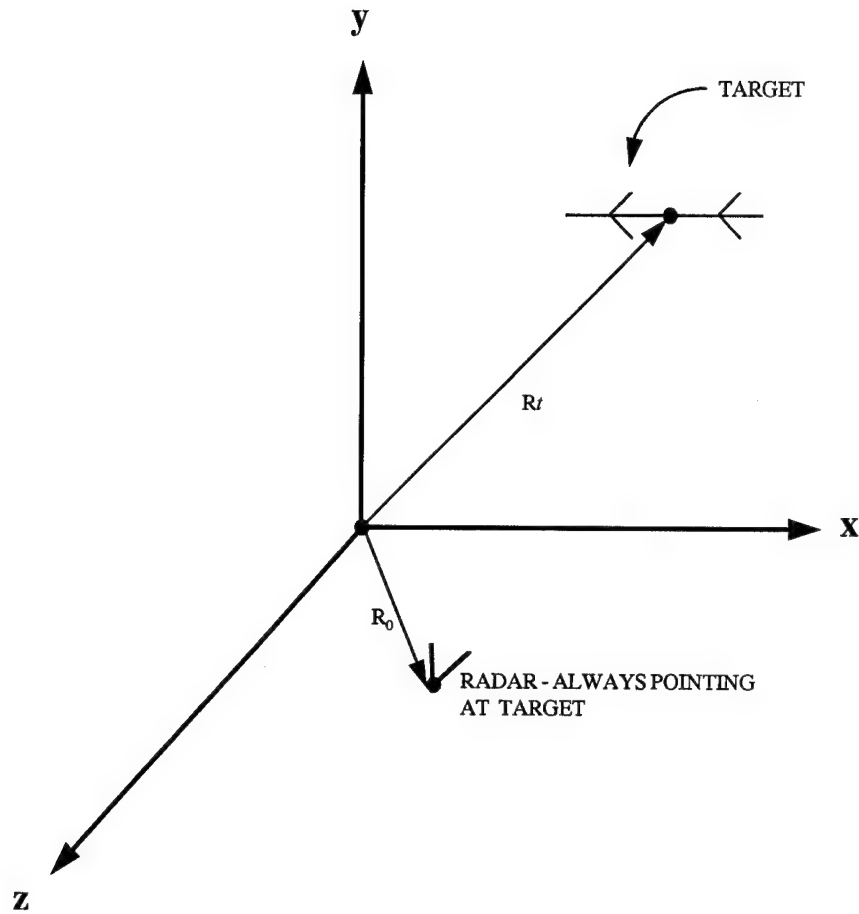


Figure 2.1: Radar Target Geometry.

and the outputs are calculated again, and so on. Also, the normalized time variable, n , increments as $n = 1, 2, \dots, ti$; where there are ti time steps in a simulation. Thus, the actual time t , in seconds, is given by $t = (n - 1)dt$.

The MATLAB code which runs the entire simulation and calls all of the MATLAB functions is called **profile.m**. This is a MATLAB script file and is commented to show where the various functions are performed.

In the simplest case, the target is flying straight and level with no perturbations (vibrations) in position or orientation. In reality this is not true. The target has perturbations in both position (translational) and orientation (rotational) from the intended flight path. Although these perturbations are very small relative to the size of the target, they are the driving factor behind angle error (glint) and Doppler

Table 2.1: Simulation Variables.

Variable	Description	Units
f	radar frequency	hertz
P_{tr}	transmitter power	watts
G_{tr}	transmitter antenna gain	numeric
V_t	target velocity	meters per second
\widehat{D}_t	target direction vector (x,y,z)	normalized
\widehat{P}_t	target pointing vector (x,y,z)	normalized
\vec{R}_t	target position vector (x,y,z)	meters
\vec{R}_o	radar position vector (x,y,z)	meters
dt	time step for simulations	seconds
ti	number of iterations	integer
t	time	seconds
n	normalized time	integer
AZ	azimuth angle	degrees
EL	elevation angle	degrees

shift. Thus, these perturbations, or vibrations, are modeled here and described in Section 2.5. Note, as stated before, the target can be any flying object, but it may also be referred to as the aircraft since the aerodynamic models assume an aircraft.

The scattering models developed in the next section apply to both the far-field and the near-field scattering from a target. Thus, the simulations developed can be applied to large radar-target range ($\vec{R}_t - \vec{R}_o$) scenarios (far-field) or for small radar-target range (near-field or end game) scenarios. Also, in the course of a simulation, both far-field and near-field situations can exist, and the simulations can accurately predict the scattered field, Doppler shift, and angular error throughout this scenario.

Next, in Section 2.2, the aircraft scattering model will be described. This simple

mathematical model calculates the scattered field from any radar target, such as an aircraft, as a function of polarization, frequency, and aspect angle for a static situation (stationary radar and target).

2.2 Aircraft Scattering Model

In order to calculate the scattered field the radar receives, the mono-static radar scattering characteristics of the aircraft must be modeled. This section describes how the static (no-motion) scattering response of a target is calculated. The model used relies on the scattering center approach to scattering, which assumes that the high frequency scattering from an electrically large target such as an aircraft can be approximated by a sum of scattering from a finite number of predominant scattering centers [5, 6, 7, 2]. The single-polarization scattering from a complicated target is thus approximated as

$$\mathbf{E}_s(f, \theta) = \mathbf{S}(f, \theta) \mathbf{E}_t(f, \theta) \left(\frac{e^{-jk_o 2r}}{r^2} \right), \quad (2.1)$$

where $\mathbf{S}(f, \theta)$ is the scattering coefficient for the entire target and can be further defined as

$$\mathbf{S}(f, \theta) = \left(\sum_{\gamma=1}^{\Gamma} \mathbf{s}_{\gamma}(f, \theta) \right), \quad (2.2)$$

where $\mathbf{s}_{\gamma}(f, \theta)$ is the scattering coefficient for the γ -th scattering center, $\mathbf{E}_s(f, \theta)$ is the scattered electric field (frequency domain) as seen at the radar receiver, $\mathbf{E}_t(f, \theta)$ is the electric field transmitted from the radar, and r is the distance from the radar to the aircraft (target) phase center. The scattering coefficient, and the scattered fields are all complex numbers and also functions of angle and frequency. They are considered frequency domain representations of the electric fields.

The term aircraft phase center refers to a point on or near the physical target center to which all of the phases of $\mathbf{s}_{\gamma}(f, \theta)$ are referenced. Then r is the distance from this point to the radar receive antenna. The angle θ is a generic angle which

defines radar-target orientation as referenced from the target. However, for all of the simulations in this report, θ is the azimuth angle (AZ) relative to the aircraft, as shown in Figure 2.2. The frequency f is the frequency of the CW electric field transmitted by the radar.

Also, this is considered a single polarization version of scattering, meaning that the polarizations of the electric fields are user definable, but fixed for a specific model. In all of the simulations, the hh polarization (horizontally polarized transmit and receive antenna on the radar) is chosen, and the C-29 transport aircraft is modeled. If different transmit or receive polarizations are required, then a different scattering coefficient, $S(f, \theta)$, will result. In the bi-static case (different transmit and receive antenna locations), then a different scattering coefficient must be used.

For all the simulations performed here, the phase of the transmitted field is assumed to be zero at the transmit antenna, while the magnitude is determined by the radar power, P_{tr} , and the radar antenna gain G_{tr} . It is assumed in all simulations that the radar antenna is bore-sighted directly at the target. Further, it is assumed that the transmission media between the target and the radar is lossless, isotropic free space. Thus, the propagation losses between the radar and the target (two-way) are taken into account by the $\left(\frac{e^{-jk_0 2r}}{r^2}\right)$ term.

The Γ scattering coefficients, $s_\gamma(f, \theta)$, contain all the target information. In this work, the scattering coefficients, which are functions of frequency (f) and radar-target aspect angle (θ), are modeled with a 2-D Damped Exponential Model. The 2-D Damped Exponential Model is given by

$$\mathbf{d}(m, n) = \sum_{\gamma=1}^{\Gamma} \mathbf{a}_\gamma \mathbf{p}_{\mathbf{x}_\gamma}^m \mathbf{p}_{\mathbf{y}_\gamma}^n \quad (2.3)$$

where

$\mathbf{p}_{\mathbf{x}_\gamma}$ = γ th x -pole, x -component of 2-D exponential (complex number)

$\mathbf{p}_{\mathbf{y}_\gamma}$ = γ th y -pole, y -component of 2-D exponential (complex number)

a_γ = γ th amplitude coefficient (complex number)

Γ = number of scattering centers (integer)

m = $0, 1, \dots, M - 1$; (normalized frequency)

n = $0, 1, \dots, N - 1$; (normalized angle).

This is a 2-D extension of Prony's Model [2]. This model assumes the radar target is comprised of Γ scattering centers. The γ -th scattering center's scattering behavior, for a single transmit and receive polarization, is modeled by the x -pole, y -pole, and amplitude coefficient triple, $\{p_{x_\gamma}, p_{y_\gamma}, a_\gamma\}$.

It is emphasized that this is a 2-D frequency domain model. A 2-D Inverse Fourier Transform of data in this domain yields data in the ISAR Image domain (or range domain). See Appendix A for a detailed description of this relationship and how an ISAR image is formed. In this report, only ISAR image plots are shown, not plots of data in the 2-D frequency domain, as plots of data in the frequency domain contain very little useful visual information.

The magnitude of the x -component damped exponential, $|p_{x_\gamma}|$, determines the dispersion of the γ -th scattering center in the x -coordinate of the transform domain (down-range in the ISAR image). By dispersion, it is meant how wide the response of the γ -th scattering center is in the down-range direction of the ISAR image (similar to a point spread function [8]). The angle of the x -component damped exponential, $\angle p_{x_\gamma}$, determines the location of the γ -th scattering center in the down-range direction. Similarly, the magnitude of the y -component damped exponential, $|p_{y_\gamma}|$, determines the dispersion of the γ -th scattering center in the y -coordinate of the transform domain (cross-range in the ISAR image) while the angle of the y -component damped exponential, $\angle p_{y_\gamma}$, determines the location of the γ -th scattering center in the cross-range direction of the ISAR image. For this work, the x -poles correspond to the down-range dimension and the y -poles correspond to the cross-range dimension in an ISAR image (see Figure 2.5).

Relating the 2-D Damped Exponential Model, Equation (2.3), to the model in Equation (2.1), we can see that

$$\mathbf{S}(f, \theta) = \mathbf{d}(m, n), \quad (2.4)$$

or

$$\sum_{\gamma=1}^{\Gamma} s_{\gamma}(f, \theta) = \sum_{\gamma=1}^{\Gamma} a_{\gamma} p_{\mathbf{x}_{\gamma}}^m p_{\mathbf{y}_{\gamma}}^n, \quad (2.5)$$

and that

$$s_{\gamma}(f, \theta) = a_{\gamma} p_{\mathbf{x}_{\gamma}}^m p_{\mathbf{y}_{\gamma}}^n \quad (2.6)$$

for $\gamma = 1, 2, \dots, \Gamma$. It must be noted that the normalized frequency m is directly related to the actual frequency f , and the normalized angle n , is directly related to the aspect angle to the aircraft θ (see Equation (2.11)). The angle θ is directly related to the azimuth and elevation angles at which the aircraft is being illuminated from. For all of the C-29 data,

$$\theta = \text{AZ}.$$

Thus, all of the C-29 data are azimuth cuts consistent with the definition of AZ shown in Figure 2.2. Note that the definition of AZ in the figure is negative azimuth (-AZ). Positive AZ values go in the clockwise direction looking from above the aircraft. Also shown in Figure 2.2 is the target direction vector, \widehat{Pt} , as defined in Table 2.1.

From the geometry of the situation, the values of AZ and EL from the aircraft to the radar can be determined. From these angles, values of m and n can be found. The values of m and n depend upon how the values of $\{p_{\mathbf{x}_{\gamma}}, p_{\mathbf{y}_{\gamma}}, a_{\gamma}\}$ were originally determined.

Consider samples of the scattering coefficient $\mathbf{S}(f, \theta)$ taken at frequencies $f = f_0, f_1, \dots, f_{M-1}$ and angles $\theta = \theta_0, \theta_1, \dots, \theta_{N-1}$. It is assumed that the frequencies and angles are evenly spaced at intervals of df and da , respectively. If the data is

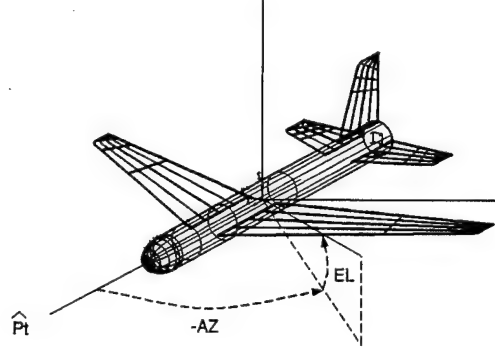


Figure 2.2: Definition of azimuth (AZ) and elevation (EL) angles for the target. The AZ angle shown is actually negative AZ (-AZ). Note that for all the C-29 data, $\theta = \text{AZ}$. Also note that \hat{Pt} is the direction the aircraft is pointing in.

formed into a matrix, an $M \times N$ data matrix called D results, and is given in by

$$D = \begin{bmatrix} S(f_0, \theta_0) & S(f_0, \theta_1) & \cdots & S(f_0, \theta_{N-1}) \\ S(f_1, \theta_0) & S(f_1, \theta_1) & \cdots & S(f_1, \theta_{N-1}) \\ \vdots & \vdots & & \vdots \\ S(f_{M-1}, \theta_0) & S(f_{M-1}, \theta_1) & \cdots & S(f_{M-1}, \theta_{N-1}) \end{bmatrix}. \quad (2.7)$$

The total frequency and angular bandwidths of the data are defined by

$$\Delta F = f_{M-1} - f_0 \quad (2.8)$$

$$\Delta \Theta = \theta_{N-1} - \theta_0 \quad (2.9)$$

As stated before, from this data, an Inverse Synthetic Aperture Radar (ISAR) Image can be formed using the 2-D Inverse Fourier Transform. For a brief description of radar imaging, please see Appendix A or consult [8, 9].

From this raw radar data contained in the matrix D , the 2-D Damped Exponential Parameters, $\{\mathbf{p}_{\mathbf{x}_\gamma}, \mathbf{p}_{\mathbf{y}_\gamma}, \mathbf{a}_\gamma\}_{\gamma=1}^T$, are estimated using the 2-D TLS-Prony Technique. This technique, described in [3], takes the raw radar data and finds the “best” parameters in a Total Least Squares sense. Once these parameters are found, they can be used over and over again to predict the scattering from a target for a range of frequencies and angles. The raw ISAR data in D is considered “unfocused” and is

used directly, without any focusing (see Appendix B), to estimate the 2-D Damped Exponential parameters.

It must be noted that the model, described by $\{\mathbf{p}_{\mathbf{x}_\gamma}, \mathbf{p}_{\mathbf{y}_\gamma}, \mathbf{a}_\gamma\}_{\gamma=1}^{\Gamma}$, is only valid over a range of m and n related to the 2-D angular and frequency range of the original raw ISAR data, ΔF and $\Delta \Theta$, from which the model was developed. Using the model outside of these limits is warranted, but only for a limited number of frequencies and angles. A good rule of thumb is to use the 2-D Damped Exponential Model at frequencies between $(f_0 - \Delta F)$ and $(f_{M-1} + \Delta F)$ and angles between $(\theta_0 - \Delta \Theta)$ and $(\theta_{N-1} + \Delta \Theta)$. This corresponds to values of normalized frequency m between $(-M+1)$ and $(2M-1)$ and values of normalized angle n between $(-\frac{N}{2})$ and $(\frac{3N-1}{2})$. For more information on this topic, which is called data extrapolation, see [10]. Also, the normalized frequency and angular variables, m and n , are related to the actual variables, f and θ , as

$$\begin{aligned} m &= \frac{f - f_0}{df} \\ n &= \frac{\theta - \theta_0}{da} + N/2. \end{aligned} \tag{2.10}$$

Now that the modeling procedure has been described, an example of how it was applied to some actual static radar data is shown. For a more detailed explanation of the 2-D Damped Exponential Model and its application to radar modeling, please consult [1, 2, 3, 11].

2.3 Damped Exponential Modeling of C-29 Scale Model

The MATLAB code which calculates the 2-D Damped Exponential parameters for the C-29 data is called **RUN.m**. The MATLAB code which contains the model parameters for the C-29 aircraft is called **C29model.m**. The raw C-29 data was obtained from a compact Inverse Synthetic Aperture Radar (ISAR) range, and the parameters of the raw ISAR data are shown in Table 2.2. The C-29 aircraft measured was a high quality one-third scale model of the actual C-29 aircraft. An outline of

the scale model is given in Figure 2.3. For more information on the C-29 aircraft consult JANE's All the World's Aircraft or [12]. The one-third scale C-29 model was completely coated with a perfect conductor. Note that there is not complete spherical coverage for the C-29 ISAR data. The scattering data only exists for limited angular coverage, and this is depicted in Figure 2.4. Also, the raw ISAR data is given as samples of the scattering coefficients, as defined in Equation (2.7).

The models derived from the raw ISAR data did not utilize all of the data, and the parameters of the data used to develop the models in **C29model.m** are shown in Table 2.3. It was assumed for all the C-29 models, that ten scattering centers would be used to model the scattering data, thus $\Gamma = 10$. As an example, lets consider Model Number 1. For Model Number 1, the 2-D Damped Exponential Parameters, determined by the 2-D TLS-Prony Technique [3], are given in Equations (2.11), (2.12), and (2.13). In these equations, the angular units are radians in the exponential arguments, and the angles range from $-\pi$ to π . Also, in the MATLAB code **C29model.m**, the C-29 models have more precision than five digits as shown in the equations.

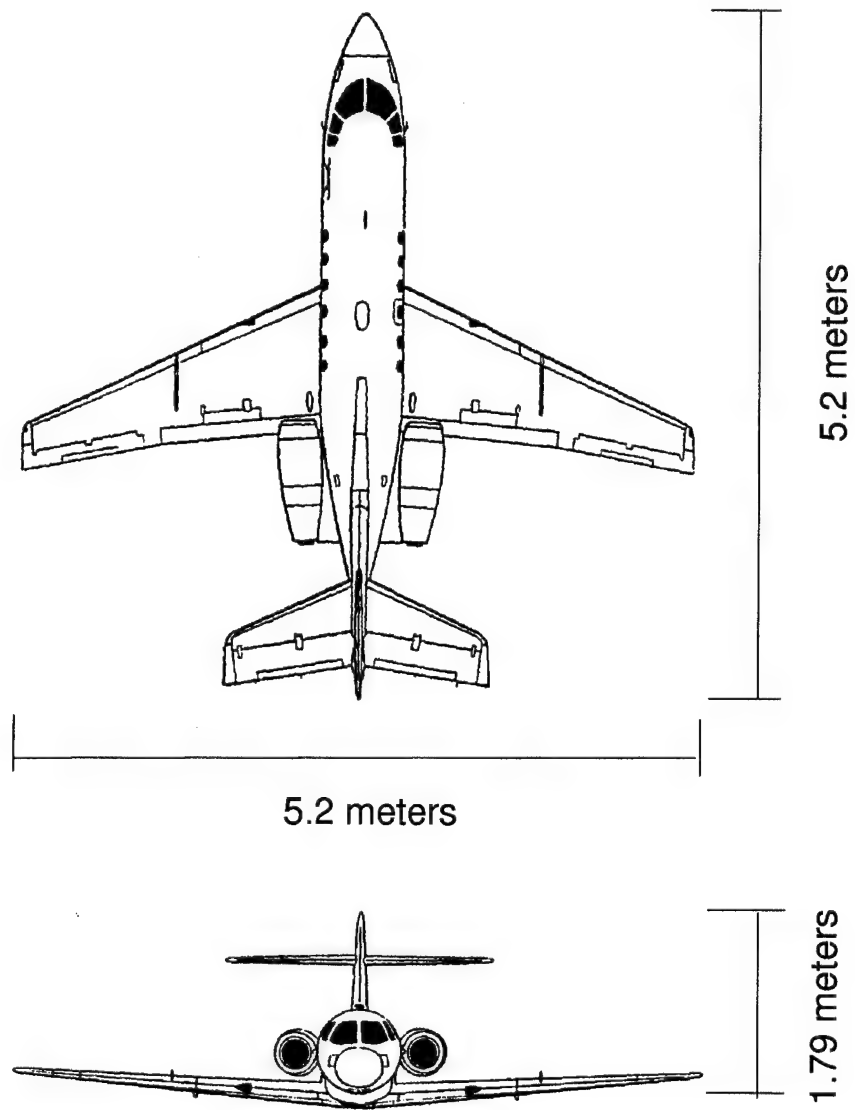


Figure 2.3: Sketch and Dimensions of C-29 one-third scale model.



Figure 2.4: Limits of angular coverage for raw C-29 ISAR data.

Table 2.2: C-29 ISAR Data Specifications, Unscaled.

File Name	Freq (GHz) Range, Step	AZ (Deg) Range, Step	EL (Deg)
C29_3D_hhN.02700	26 to 36, 0.01	-5 to 5, 0.04	3
C29_3D_hhN.02500	26 to 36, 0.01	-5 to 5, 0.04	5
C29_3D_hhN.02300	26 to 36, 0.01	-5 to 5, 0.04	7
C29_3D_hhW.02700	26 to 36, 0.01	19 to 29, 0.04	3
C29_3D_hhW.02500	26 to 36, 0.01	19 to 29, 0.04	5
C29_3D_hhW.02300	26 to 36, 0.01	19 to 29, 0.04	7
C29_3D_hhW.02700	26 to 36, 0.01	-19 to -29, 0.04	3
C29_3D_hhW.02500	26 to 36, 0.01	-19 to -29, 0.04	5
C29_3D_hhW.02300	26 to 36, 0.01	-19 to -29, 0.04	7

Table 2.3: **C29model.m** Data Specifications, Scaled

Model No.	Freq Range, Steps (GHz)	Center AZ, Steps (Deg)	EL (Deg)	($M \times N$)
1	8.67 to 9.09, 0.0067	-3.76, 0.04	3	64×64
2	8.67 to 9.09, 0.0067	-0.08, 0.04	3	64×64
3	8.67 to 9.09, 0.0067	3.72, 0.04	3	64×64
4	8.67 to 9.09, 0.0067	-3.76, 0.04	5	64×64
5	8.67 to 9.09, 0.0067	-0.08, 0.04	5	64×64
6	8.67 to 9.09, 0.0067	3.72, 0.04	5	64×64
7	8.67 to 9.09, 0.0067	-3.76, 0.04	7	64×64
8	8.67 to 9.09, 0.0067	-0.08, 0.04	7	64×64
9	8.67 to 9.09, 0.0067	3.72, 0.04	7	64×64
10	8.67 to 9.09, 0.0067	20.24, 0.04	3	64×64
11	8.67 to 9.09, 0.0067	23.92, 0.04	3	64×64
12	8.67 to 9.09, 0.0067	27.72, 0.04	3	64×64
13	8.67 to 9.09, 0.0067	20.24, 0.04	5	64×64
14	8.67 to 9.09, 0.0067	23.92, 0.04	5	64×64
15	8.67 to 9.09, 0.0067	27.72, 0.04	5	64×64
16	8.67 to 9.09, 0.0067	20.24, 0.04	7	64×64
17	8.67 to 9.09, 0.0067	23.92, 0.04	7	64×64
18	8.67 to 9.09, 0.0067	27.72, 0.04	7	64×64
19	8.67 to 9.09, 0.0067	-20.24, 0.04	3	64×64
20	8.67 to 9.09, 0.0067	-23.92, 0.04	3	64×64
21	8.67 to 9.09, 0.0067	-27.72, 0.04	3	64×64
22	8.67 to 9.09, 0.0067	-20.24, 0.04	5	64×64
23	8.67 to 9.09, 0.0067	-23.92, 0.04	5	64×64
24	8.67 to 9.09, 0.0067	-27.72, 0.04	5	64×64
25	8.67 to 9.09, 0.0067	-20.24, 0.04	7	64×64
26	8.67 to 9.09, 0.0067	-23.92, 0.04	7	64×64
27	8.67 to 9.09, 0.0067	-27.72, 0.04	7	64×64

$$\mathbf{a} = \begin{bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \\ \mathbf{a}_3 \\ \mathbf{a}_4 \\ \mathbf{a}_5 \\ \mathbf{a}_6 \\ \mathbf{a}_7 \\ \mathbf{a}_8 \\ \mathbf{a}_9 \\ \mathbf{a}_{10} \end{bmatrix} = \begin{bmatrix} -0.1187 + 0.1311j \\ 0.0016 + 0.0009j \\ 0.0001 - 0.0000j \\ 0.1069 + 0.1440j \\ -0.0002 - 0.1779j \\ 0.0008 + 0.0812j \\ -0.0316 - 0.0104j \\ -0.0701 + 0.0726j \\ -0.0108 - 0.0978j \\ -0.0694 + 0.0745j \end{bmatrix} = \begin{bmatrix} 0.1768e^{j(2.3067)} \\ 0.0018e^{j(0.5195)} \\ 0.0001e^{j(-0.2738)} \\ 0.1793e^{j(0.9321)} \\ 0.1779e^{j(-1.5717)} \\ 0.0812e^{j(1.5613)} \\ 0.0332e^{j(-2.8244)} \\ 0.1009e^{j(2.3385)} \\ 0.0984e^{j(-1.6806)} \\ 0.1018e^{j(2.3207)} \end{bmatrix} \quad (2.11)$$

$$\mathbf{p}_x = \begin{bmatrix} \mathbf{p}_{x_1} \\ \mathbf{p}_{x_2} \\ \mathbf{p}_{x_3} \\ \mathbf{p}_{x_4} \\ \mathbf{p}_{x_5} \\ \mathbf{p}_{x_6} \\ \mathbf{p}_{x_7} \\ \mathbf{p}_{x_8} \\ \mathbf{p}_{x_9} \\ \mathbf{p}_{x_{10}} \end{bmatrix} = \begin{bmatrix} 1.0073 - 0.1202j \\ 0.3435 - 1.0489j \\ 0.1962 - 1.1428j \\ 0.9399 - 0.3710j \\ 0.9856 + 0.2002j \\ 0.8361 - 0.5852j \\ -0.1659 - 1.0266j \\ 1.0073 - 0.1202j \\ 0.9399 - 0.3710j \\ -0.1206 + 0.9831j \end{bmatrix} = \begin{bmatrix} 1.0145e^{j(-0.1188)} \\ 1.1037e^{j(-1.2544)} \\ 1.1595e^{j(-1.4008)} \\ 1.0104e^{j(-0.3759)} \\ 1.0057e^{j(0.2004)} \\ 1.0205e^{j(-0.6107)} \\ 1.0399e^{j(-1.7310)} \\ 1.0145e^{j(-0.1188)} \\ 1.0104e^{j(-0.3759)} \\ 0.9905e^{j(1.6928)} \end{bmatrix} \quad (2.12)$$

$$\mathbf{p}_y = \begin{bmatrix} \mathbf{p}_{y_1} \\ \mathbf{p}_{y_2} \\ \mathbf{p}_{y_3} \\ \mathbf{p}_{y_4} \\ \mathbf{p}_{y_5} \\ \mathbf{p}_{y_6} \\ \mathbf{p}_{y_7} \\ \mathbf{p}_{y_8} \\ \mathbf{p}_{y_9} \\ \mathbf{p}_{y_{10}} \end{bmatrix} = \begin{bmatrix} 0.9246 - 0.3891j \\ 1.0017 + 0.0050j \\ 1.0017 + 0.0050j \\ 1.0017 + 0.0050j \\ 1.0017 + 0.0050j \\ 0.9246 - 0.3891j \\ 1.0017 + 0.0050j \\ 1.0017 + 0.0050j \\ 0.9542 - 0.2781j \\ 0.9577 + 0.3217j \end{bmatrix} = \begin{bmatrix} 1.0031e^{(-0.3983)} \\ 1.0017e^{(0.0049)} \\ 1.0017e^{(0.0049)} \\ 1.0017e^{(0.0049)} \\ 1.0017e^{(0.0049)} \\ 1.0031e^{(-0.3983)} \\ 1.0017e^{(0.0049)} \\ 1.0017e^{(0.0049)} \\ 0.9939e^{(-0.2836)} \\ 1.0103e^{(0.3241)} \end{bmatrix} \quad (2.13)$$

To calculate the scattering coefficient, $\mathbf{S}(f, \theta) = \mathbf{d}(m, n)$, at $EL = 3^\circ$, $AZ = -4^\circ$, and $f = 9GHz$, we use Equations (2.3) and (2.11), to obtain

$$m = 50.0 \quad (2.14)$$

$$n = 26.0 \quad (2.15)$$

and thus

$$\mathbf{S}(9GHz, 3^\circ) = \mathbf{d}(50, 26) = 0.3577 + 0.3041j = 0.4695e^{j0.7047}. \quad (2.16)$$

This example shows how to calculate the scattering coefficient for the target at a given frequency and aspect angle. The angular variable θ used here directly corresponds to AZ angle for the C-29. To get a better appreciation for the 2-D damped exponential model, consider Model Number 1 in Table 2.3. Using the raw ISAR data used to calculate Model Number 1 and the 2-D Inverse Fast Fourier Transform (IFFT), an ISAR image was formed (see Appendix A.) and is shown in Figure 2.5. Note that this image was formed from actual ISAR data, not the 2-D Damped Exponential Model. Also note that an outline of the C-29 Aircraft is shown on the plot

to orient the reader as to where the scattering is originating from on the C-29 scale model. The dimensions of the image plots have been scaled to full-scale C-29 size.

Next, the 2-D Damped Exponential Model, Model Number 1, was used to generate ISAR data over the same angular and frequency ranges as C-29 Model Number 1, and a model based image was formed. This image is shown in Figure 2.6. As before, an overlay of the C-29 is shown. Also, the scattering center locations of the ten scattering centers defined by the angles of \mathbf{p}_x and \mathbf{p}_y , as given in Equations (2.12) and (2.13), are overlayed on the two images and denoted by 'o's. Comparing Figures 2.5 and 2.6, it can be seen the 2-D Damped Exponential Model does not reproduce an exact image, but does reproduce an image which is visually close to the actual ISAR image. Also, the scattering centers are placed on the ISAR image at locations where most of the scattering energy is being reflected from.

Using the 2-D Damped Exponential (2-D Prony's) Model, many different radar targets can be simulated by placing scattering centers in various locations. The MATLAB code `point_model.m` creates a generic aircraft by placing scattering centers at probable scattering locations. This "point_model" can be used by `profile.m` to generate an entire radar-target engagement. Now that the scattering characteristic of the target have been described, the scattered field can easily be calculated, as discussed in the next section.

2.4 Scattered Field Calculations

To calculate the scattered field at time t , we can use the variables in Table 2.1 along with Equations (2.1) and (2.4). First, define the distance from the target phase center to the radar receive antenna as

$$r = |\vec{R}t - \vec{R}o| \quad (2.17)$$

where $|\cdot|$ stands for spatial distance in this case. Using the definitions of P_{tr} and

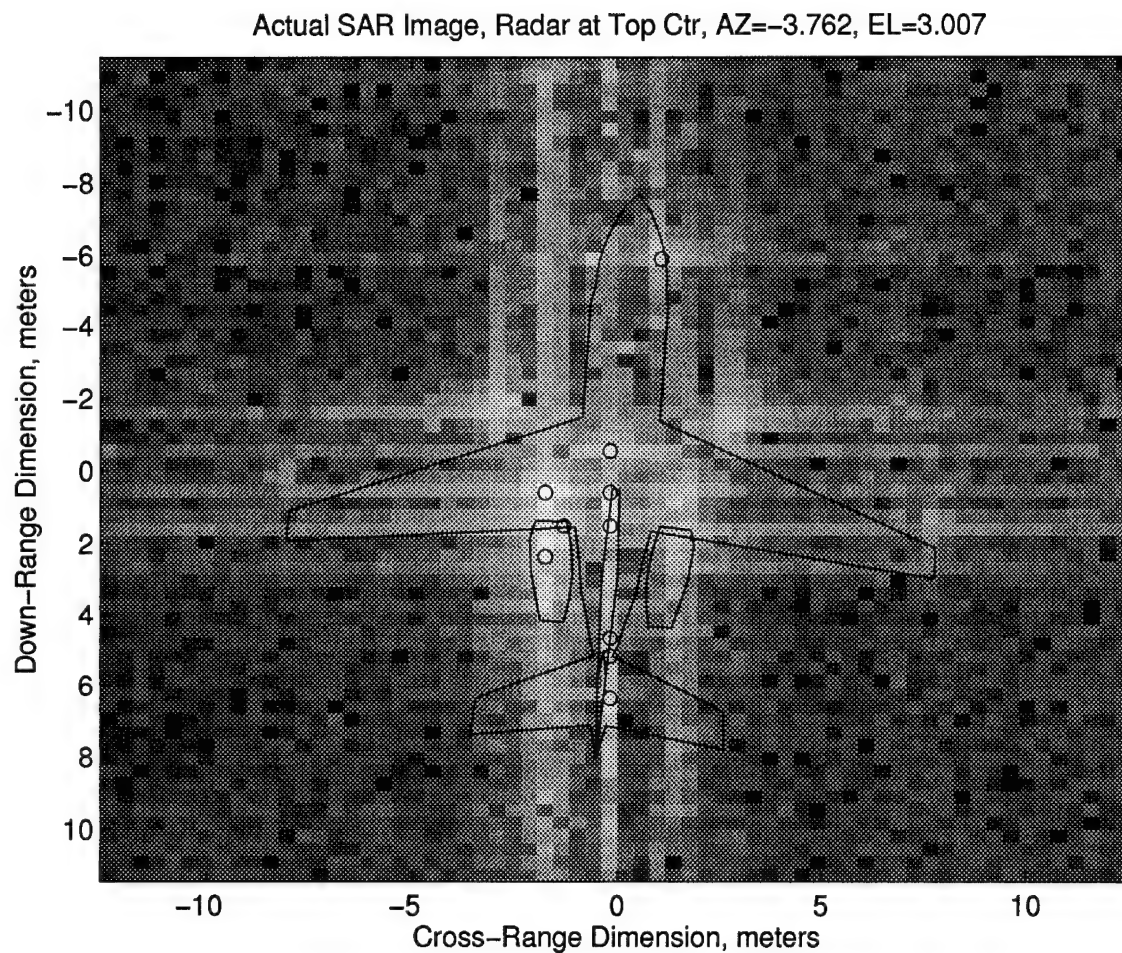


Figure 2.5: Actual ISAR Image for One-Third C-29 Scale Model. An overlay of the C-29 Aircraft is shown. Dimensions have been scaled to correspond to a full-size C-29 Aircraft.

Model Gen SAR Image, Radar at Top Ctr, AZ=-3.762, EL=3.007

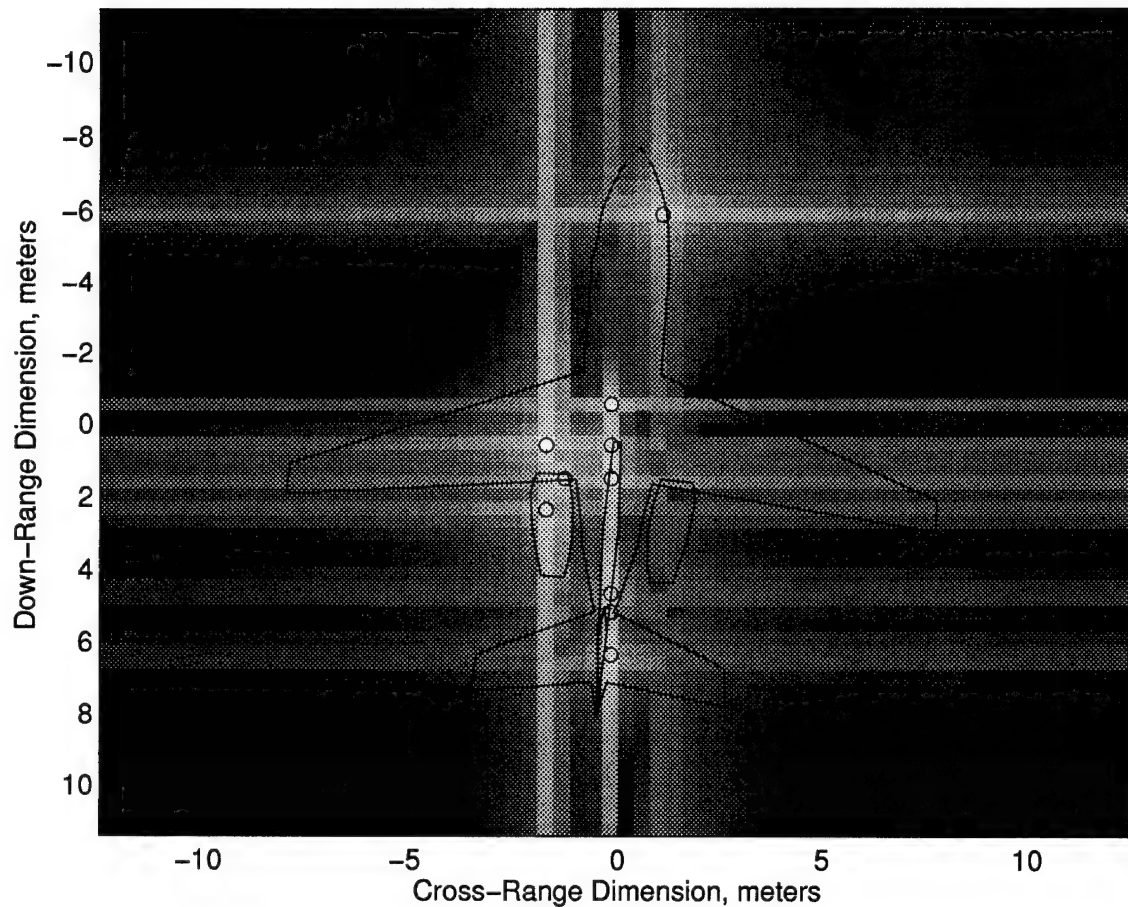


Figure 2.6: Model Generated ISAR Image for One-Third C-29 Scale Model. An overlay of the C-29 Aircraft is shown. Dimensions have been scaled to correspond to a full size C-29 Aircraft.

Gtr , the complex frequency domain value of the scattered field as seen by the receiver is

$$\mathbf{E}_{\text{scat}} = d(m, n) \left\{ \frac{2(Ptr)(Gtr)\eta_o}{4\pi r^2} \right\} e^{-jk_o 2r} \quad (2.18)$$

where

$$k_o = \frac{2\pi f}{c} \quad (2.19)$$

$$\eta_o = 376.7 \ \Omega$$

$$c = 2.9979 \times 10^8 \text{ m/s.}$$

This equation was derived assuming a mono-static radar, and that the velocity of the aircraft was much smaller than the speed of light (which is a pretty good assumption). From this complex quantity, an actual time domain expression for the scattered field can be found. Let

$$\mathbf{E}_{\text{scat}} = |\mathbf{E}_{\text{scat}}| e^{j\angle \mathbf{E}_{\text{scat}}}, \quad (2.20)$$

where $|\cdot|$ denotes complex magnitude and \angle denotes complex angle. Next, the $e^{j\omega t}$ dependence is added back into the frequency domain expression to yield

$$\mathbf{E}_{\text{scat}}^t = \mathbf{E}_{\text{scat}} e^{j\omega t}. \quad (2.21)$$

Next, define

$$E_o = |\mathbf{E}_{\text{scat}}^t| \quad (2.22)$$

$$\phi_o = \angle \mathbf{E}_{\text{scat}}^t \quad (2.23)$$

where E_o and ϕ_o are the magnitude and phase of the complex number $\mathbf{E}_{\text{scat}}^t$. Finally, a time domain expression for the scattered field is given by

$$E_s(t) = E_o \cos(\omega t + \angle \mathbf{E}_{\text{scat}}) = E_o \cos(\phi_o). \quad (2.24)$$

This time domain expression is valid for time t , frequency f ($\omega = 2\pi f$), and the physical situation described above. After the target and/or the radar move, \mathbf{E}_{scat} will change its value and thus change the values $\mathbf{E}_{\text{scat}}^t$ and $E_s(t)$.

Some important assumptions have been made in order to calculate the scattered field in Equation (2.24). First, the 2-D Damped Exponential Model is physically, by definition, a two-dimensional model. Thus, the aircraft is modeled in space by a finite plate, or plane in 3-D space. The direction of this plane is defined by the directions $\hat{P}t$ and a line parallel to the xz -plane, as demonstrated in Figure 2.1. For the simple case of $\hat{P}t$ lying in the $(-x)$ -direction, the aircraft model is parallel to the xz -plane. Second, all of the scattering centers lie in this plane within the physical boundaries of the unambiguous range of the target. The unambiguous range is defined by the frequency sampling rate of the raw ISAR data, df and da . The unambiguous range in the down-range dimension is

$$Ru_f = \frac{c}{2(df)} \quad (2.25)$$

while the unambiguous range in the cross-range dimension of the image is

$$Ru_a = \frac{c}{2(df a)} \quad (2.26)$$

where $dfa = (da)_{\text{rad}} f$ (da_{rad} is the value of da in radians). The down-range and cross-range unambiguous ranges are determined by Nyquist's sampling criteria on the raw ISAR data in the 2-D frequency domain. Also note that in the ISAR images shown in Figure 2.5, the unambiguous ranges are the limits on the horizontal (cross-range) and vertical (down-range) axes. If the actual target size exceeds the unambiguous ranges, aliasing will occur in the ISAR image and 2-D Damped-Exponential parameters can not be estimated.

Now that the scattering characteristics of the target are known, aircraft motions and vibrations can be added to the nominal flight path to better simulate a dynamic aircraft in flight.

2.5 Aircraft Vibrations and Motion Models

It is well known that any aircraft vibrates slightly during flight. Actually, there are many modes of vibrations. These vibrations are significant factors with respect to the Doppler shift and angular error as seen by a radar tracking a target. There are two types of aircraft motion simulated here. The first is called rigid-body motion, and the second is called individual motion. Rigid-body motion refers to the entire aircraft vibrating, or moving, away from the intended path of flight as a rigid-body. In this case there is no flexing of the aircraft at all. The second case involves flexing of the aircraft. In this case the aircraft flexes and the various parts of the aircraft move with respect to one another. Next, each type of motion is described.

2.5.1 Rigid Body Motion

In rigid body motion, there is no flexing of the aircraft structure at all. There are two types of rigid body motion, translational and rotational. For translational motion, the entire aircraft translates from its nominal flight path in all three Cartesian directions at each instant of time. For rotational motion, the entire aircraft rotates from its direction in all three angular directions (roll, pitch, and yaw). The MATLAB code which calculates the translational and rotational rigid-body motion is called `rigid_motion.m`. First, let's consider translational motion.

Translational Rigid Body Motion

The aircraft position with no motion is given by $\vec{R}t$ for a given time t . Translational motion changes the (x, y, z) components of $\vec{R}t$ by slight amounts to simulate the motion. Consider three independent identically distributed white Gaussian random processes, wtn_x , wtn_y , and wtn_z , each with zero mean and standard deviation σ_{rt} . Assuming there are ti time steps in the simulation, each process has ti samples. First, each process is filtered by a 16-th order FIR low-pass filter with cut-off frequency ft_c with discrete-time impulse response $h_{tr}[n]$. The resulting three processes are ctn_x ,

ctn_y , and ctn_z , defined by

$$ctn_x = wtn_x \otimes h_{tr} \quad (2.27)$$

$$ctn_y = wtn_y \otimes h_{tr} \quad (2.28)$$

$$ctn_z = wtn_z \otimes h_{tr} \quad (2.29)$$

where \otimes denotes discrete-time convolution [13]. Next a 3×3 correlation matrix is defined as

$$C_t = \begin{bmatrix} \rho_{xx} & \rho_{xy} & \rho_{xz} \\ \rho_{yx} & \rho_{yy} & \rho_{yz} \\ \rho_{zx} & \rho_{zy} & \rho_{zz} \end{bmatrix}. \quad (2.30)$$

At each instant of time, three new random variables are defined as

$$\begin{bmatrix} ct_x[n] \\ ct_y[n] \\ ct_z[n] \end{bmatrix} = C_t \begin{bmatrix} ctn_x[n] \\ ctn_y[n] \\ ctn_z[n] \end{bmatrix}. \quad (2.31)$$

Thus, the three new random processes, ct_x , ct_y , and ct_z are colored (by filtering) and correlated (by Equation (2.31)) with one another. At time t , the new aircraft position is

$$\vec{R}t_t = \vec{R}t + \begin{bmatrix} ct_x[n] \\ ct_y[n] \\ ct_z[n] \end{bmatrix}. \quad (2.32)$$

Thus, the aircraft is translationally shifted from the intended flight path by bandlimited correlated white noise. Note that the magnitude of the shifts tend to be smaller than σ_{rt} since the low-pass filtering filters out the high frequency, or large magnitude components. The cut-off frequency, ft_c , is nominally around 2 to 5 Hz. The standard deviation of the translational movement, σ_{rt} , is around 10 cm per dimension, while

the correlation matrix is typically

$$C_t = \begin{bmatrix} 0.9 & 0.1 & 0.1 \\ 0.1 & 0.9 & 0.1 \\ 0.1 & 0.1 & 0.9 \end{bmatrix}. \quad (2.33)$$

All of these values were chosen by consulting with an aeronautical engineer [14].

Rotational Rigid Body Motion

Next, the rotational rigid body motion is defined. This consists to perturbations the physical direction the aircraft is pointing. Recall the unit vector \widehat{Dt} is the direction the aircraft is traveling in while the unit vector \widehat{Pt} is the direction the aircraft is physically pointing in. See Figure 2.2 for a visual definition of \widehat{Pt} . Next, define the three orthogonal angular components, roll \mathcal{R} , pitch \mathcal{P} , and yaw \mathcal{Y} as the standard roll, pitch, and yaw angles associated with an aircraft in flight. Defining the angular vector \mathcal{A} ,

$$\mathcal{A} = \begin{bmatrix} \mathcal{R} \\ \mathcal{P} \\ \mathcal{Y} \end{bmatrix}. \quad (2.34)$$

If $\mathcal{A} = [0 \ 0 \ 0]^T$, where T denotes transpose, then roll, pitch, and yaw are all zero. In this case, the aircraft centerline is pointing in the \widehat{Dt} direction and $\widehat{Dt} = \widehat{Pt}$. When roll, pitch, and yaw are not zero, then $\mathcal{A} \neq [0 \ 0 \ 0]^T$, and $\widehat{Dt} \neq \widehat{Pt}$. In this case, the unit vector \widehat{Pt} is pointing in the direction defined by the roll, pitch, and yaw values. The aircraft is always traveling in the \widehat{Dt} direction, but is always pointing in the \widehat{Pt} direction.

Thus, \widehat{Pt} is a function of \widehat{Dt} and \mathcal{A} . When $\widehat{Dt} \neq \widehat{Pt}$, the aircraft is no longer oriented in the \widehat{Dt} direction, and new azimuth (AZ) and elevation (EL) angles need to be found (see Figure 2.2). The MATLAB code `scattered_field.m` calculates the new AZ and EL angles due to any non-zero values in \mathcal{A} . The equations for the new

AZ and EL angles are not given here, but are given in `scattered_field.m`, shown in Appendix B.

As with translation motion, rotational rigid-body motion consists of adding correlated bandlimited noise to the roll, pitch, and yaw, or to \widehat{Pt} . Since the movement values are low-pass filtered, the actual angular perturbations tend to be smaller than σ_{rr} , the standard deviation of the original white Gaussian processes for rotational rigid body motion. Defining the movement vector in the same fashion as for translational motion,

$$\vec{cr}[n] = \begin{bmatrix} cr_x[n] \\ cr_y[n] \\ cr_z[n] \end{bmatrix} \quad (2.35)$$

and the new aircraft pointing direction becomes (at time step n)

$$\widehat{Pt} = \widehat{Pt} + \vec{cr}[n]. \quad (2.36)$$

The standard deviation of the rotational movement, σ_{rr} , is typically around 2° and bandlimited to 2 to 5 Hz in each angular dimension. The correlation matrix for rotational motion is identical to the one for translational motion [14].

Translational and rotational rigid body motion are critical to the dynamic simulation of a radar-target encounter since this motion has a significant effect on scattered field values, Doppler shift, and angular error. The other type of motion considered is called individual scattering center motion. Unlike rigid-body motion, in individual scattering center motion the scattering centers on the target move independent to one another. This type of motion is described next.

2.5.2 Individual Scattering Center Motion

In order to simulate flexing of the aircraft structure, which does occur, individual movement of the scattering centers is implemented in the simulations. The scattering centers are allowed to move in the physical 2-D plane in which the scattering

centers reside. Thus, the down-range and cross-range locations of the scattering centers are varied. Assuming that there are Γ scattering centers, 2Γ discrete time, independent identically distributed Gaussian random processes are created, each of length ti . The standard deviation of all of these processes is σ_{im} . The processes are called $\{uIMdr_i\}_{i=1}^{\Gamma}$ and $\{uIMxr_i\}_{i=1}^{\Gamma}$. Next, each of the random processes are filtered through a 16-th order discrete-time low-pass FIR filter to produce the colored sequences $\{IMdr_i\}_{i=1}^{\Gamma}$ and $\{IMxr_i\}_{i=1}^{\Gamma}$. The cut-off frequency for the filter is fim_c .

Next, it is desired to change the positions of the individual scattering centers by adding the values of $IMdr_i$ and $IMxr_i$ to the positions of the individual scattering centers. Recall from Section 2.2 that the angles of the x and y -poles determine the locations of the scattering centers in the ISAR image. Thus, the angles of the x and y -poles will be individually changed in order to move the scattering centers around on the image. Also, for a given instant in time (n) and one scattering center, the down-range and cross-range movement distances are given by $IMdr_i[n]$ and $IMxr_i[n]$. However, before these perturbations are applied, the down-range and cross-range components are correlated using a 2×2 correlation matrix, yielding $cIMdr_i$ and $cIMxr_i$. Thus,

$$\begin{bmatrix} cIMdr_i \\ cIMxr_i \end{bmatrix} = \begin{bmatrix} \rho_{dd} & \rho_{dx} \\ \rho_{xd} & \rho_{xx} \end{bmatrix} \begin{bmatrix} IMdr_i \\ IMxr_i \end{bmatrix}. \quad (2.37)$$

The movement of the individual scattering centers will change the scattered field from the target. Recall that the scattered field is calculated using Equation (2.18). The value of $d(m, n)$ will change if the scattering centers move since movement of the scattering centers is simulated by changing the angles of the x and y -poles.

As an example, consider 50 overlayed realizations of individual scattering center movement as seen on the ISAR images shown in Figures 2.7 and 2.8. This corresponds to how the individual scattering centers move as the simulation progresses through 50 time increments. In Figure 2.7, the standard deviation of movement, σ_{im} , is set to 2 cm. The scattering center locations are denoted by '+'s while the original locations

are denoted by 'o's. Figure 2.8 shows the case when $\sigma_{im} = 20\text{cm}$. For both cases, $f_{im_c} = 20\text{Hz}$ and the correlation matrix is given by

$$\begin{bmatrix} \rho_{dd} & \rho_{dx} \\ \rho_{xd} & \rho_{xx} \end{bmatrix} = \begin{bmatrix} 0.8 & 0.2 \\ 0.2 & 0.8 \end{bmatrix}. \quad (2.38)$$

In each figure the model based ISAR image is shown, generated from the “un-moved” 2-D Damped Exponential parameters. An aircraft overlay is also shown with each plot.

Please note that 20cm is somewhat large for the magnitude of the individual motion, but it is shown here to demonstrate how the scattering centers change position. More realistic movement numbers are 1 to 5cm, values within this range are used for the simulations in Section 4.

The individual scattering center motion developed here is implemented in the MATLAB function `indiv_mot.m`. This type of motion is critical to achieve Doppler signatures and angular errors which are realistic, as shown in Section 4. It is important to note that the individual scattering center motion values do not depend upon where on the target the scattering center is located. Thus, a scattering center at the phase center of the target will, statistically, be moved just like a scattering center near the edge of the unambiguous range. As an upgrade to this simulation package, some weighting factor could be imposed upon the magnitude and frequency of individual scattering center motion as a function of each scattering center's location. This is discussed more in Section 6.

2.6 Summary of Dynamics and Target Scattering

In summary, the typical radar-target engagement consists of a stationary radar (which could be a missile seeker) positioned at the origin in Figure 2.1 and a target (C-29 aircraft) flying in 3-D space. As the aircraft is flying straight and level, it is vibrating both as a rigid-body and as a flexing aircraft. At discrete time intervals, dt ,

Model Gen SAR Image, Radar at Top Ctr, AZ=-3.762, EL=3.007

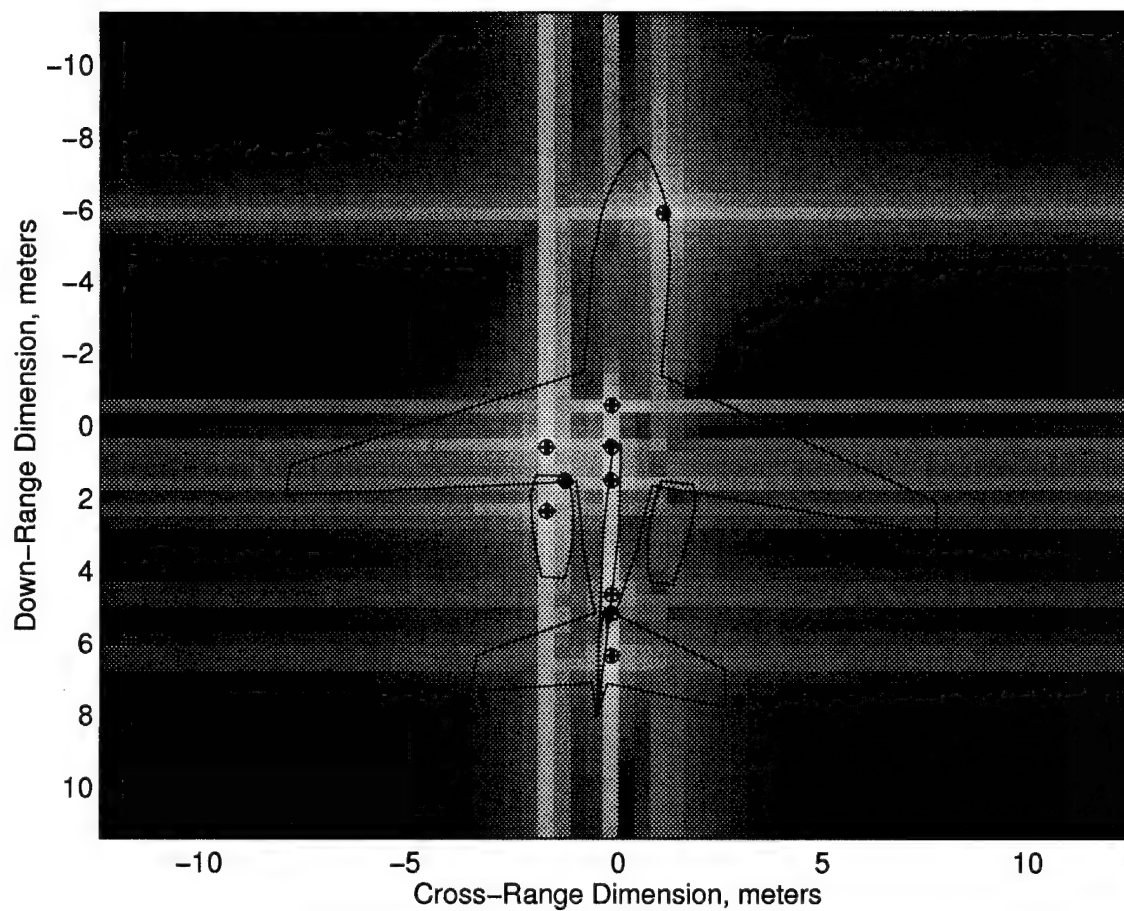


Figure 2.7: Individual Scattering Center Motion Plots. Plot shows 2cm motion with 50 overlaid realizations. An aircraft overlay is shown with the plot.

Model Gen SAR Image, Radar at Top Ctr, AZ=-3.762, EL=3.007

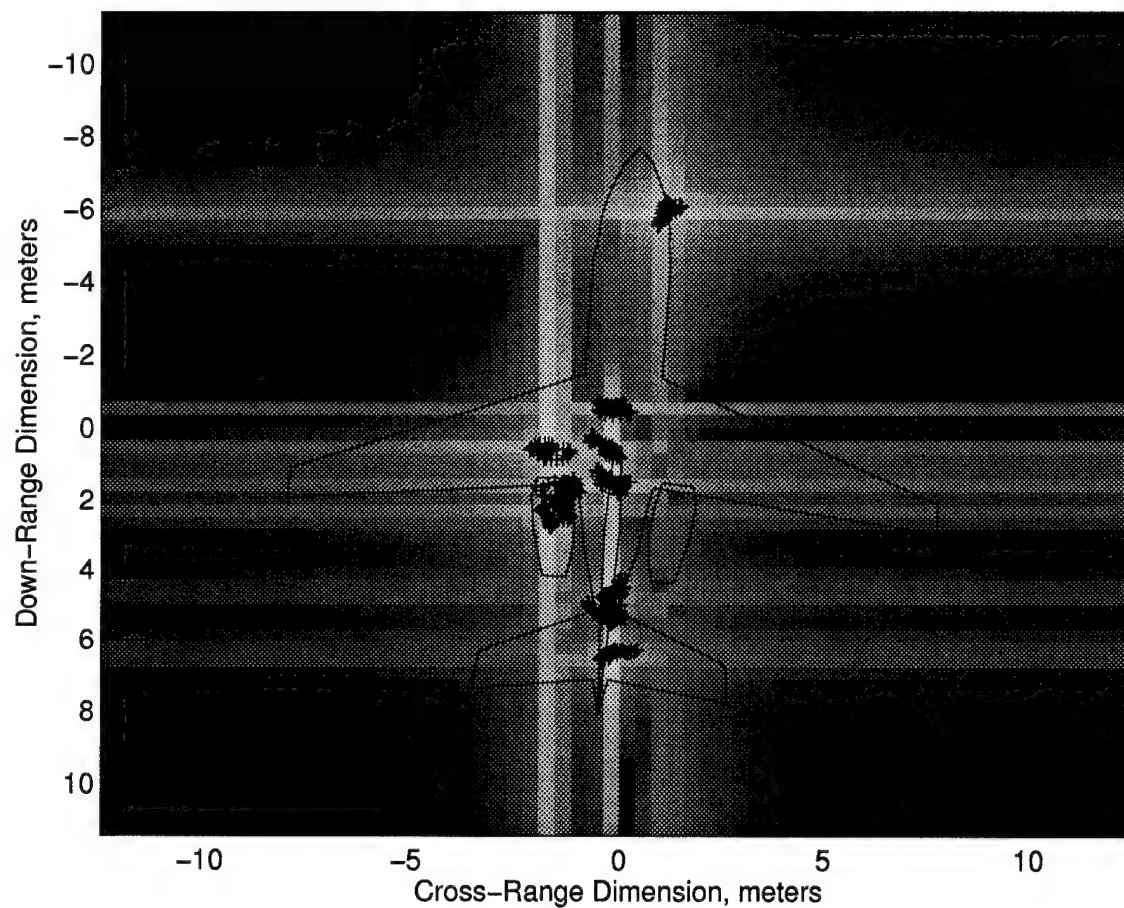


Figure 2.8: Individual Scattering Center Motion Plots. Plot shows 20cm motion with 50 overlaid realizations. An aircraft overlay is shown with the plot.

the simulation calculates the scattered electric field, the Doppler frequency shift, and the angular error (glint) all as seen at the radar. In the next, all of the simulation outputs, scattered field, Doppler shift, and angular error are described in detail.

3. Simulation Outputs

In this section, the critical output quantities of the simulations are derived and defined. They consist of the scattered electric field, the Doppler shift, and the angular error (glint) as seen at the radar receiver. Each of these quantities is calculated at each time step in the simulation.

3.1 Scattered Field

The scattered field is a critical output quantity as this is the value of the electric field as seen at the radar receiver. In a hardware in the loop simulation, this number gives the value of the electric field which should be induced at the receiver at time t . The polarization of the electric field will be defined by the polarization of the scattering coefficient in Equation (2.2). For the C-29 hh -polarization data, the scattered field (and the incident field) is horizontally polarized. The actual value of the scattered field was developed in Section 2.4, and is given by $E_s(t)$ in Equation (2.24). This value is calculated in the MATLAB code `scattered_field.m` and is passed to `profile.m`.

If the magnitude and phase of the scattered field are desired as a function of time, then they are also given in Section 2.4 as E_o and ϕ_o in Equations (2.23) and (2.23), respectively. These quantities are calculated in the MATLAB code `scattered_field.m` but are not passed to `profile.m`. They are not part of the standard output of the simulation package, but if desired can be extracted from `scattered_field.m`.

Another useful quantity related to the scattered electric field is the power received at the radar, denoted Pr . Assuming the radar receive antenna gain is the same as the transmitter antenna gain, which is appropriate for a mono-static radar, the power received immediately after the radar receive antenna is given by

$$Pr = |E_s(t)|^2 \frac{2\pi f^2}{\eta_o c^2} Gtr \quad (3.1)$$

where $|\cdot|$ denotes absolute value, η_o and c are defined in Equation (2.20), Gtr is defined in Table 2.1, and f is the frequency. This value is also calculated in the MATLAB code `scattered_field.m`.

3.2 Doppler Shift

The Doppler shift is an important quantity for radar receivers since this will define the “Doppler Bandwidth” over which a radar must operate. The Doppler shift refers to the instantaneous frequency of the scattered electric field at the radar receiver. The standard definition of Doppler shift is the difference between the frequency of the electric field with radar-target motion and that without radar-target motion. For example, if a CW radar is illuminating a target with an $f = 9$ GHz electric field, and the target is moving at 100 meters/sec directly toward the radar, then the Doppler shift would be given by [4]

$$f_d = +\frac{2(Vt)f}{c} = +6 \text{ kHz}. \quad (3.2)$$

The frequency of the scattered field received by the radar is given by

$$f_r = f + f_d. \quad (3.3)$$

For this numerical example, the received frequency would be 9.000006 GHz. This scenario is a very simple case. When both the target and radar are moving, along with different parts of the target moving at different rates (individual motion), the

calculation of the Doppler shift is a bit more involved. We will derive a general expression for the Doppler shift as follows.

Starting with Equation (2.18), we can rewrite the frequency domain expression for the scattered electric field for a given frequency f (normalized to m) and aspect angle θ (normalized to n) at time t as

$$\mathbf{E}_{\text{scat}}^t = \left\{ \sum_{i=1}^{\Gamma} \mathbf{a}_i \mathbf{p}_{\mathbf{x}_i}^m \mathbf{p}_{\mathbf{y}_i}^n \right\} \left\{ \frac{2(Ptr)(Gtr)\eta_o}{4\pi r^2} \right\} e^{-jk_o 2r} e^{j\omega t}. \quad (3.4)$$

This can also be written as

$$\mathbf{E}_{\text{scat}}^t = \sum_{i=1}^{\Gamma} E_i e^{j(\omega t + \phi_i - 2k_o \hat{n}_i \cdot \vec{R}_{ot})} \quad (3.5)$$

where

$$\Gamma = \text{number of scattering centers} \quad (3.6)$$

$$E_i = \text{amplitude of } i\text{-th scattering center} \quad (3.7)$$

$$\phi_i = \text{phase of the } i\text{-th scattering center} \quad (3.8)$$

$$\hat{n}_i = \text{unit vector pointing from } i\text{-th scattering center to radar} \quad (3.9)$$

$$\vec{R}_{ot} = \vec{R}_t - \vec{R}_o, \text{ vector from phase center of target to radar} \quad (3.10)$$

$$\omega = \text{radian frequency} (\omega = 2\pi f) \quad (3.11)$$

$$\cdot = \text{standard vector dot product.} \quad (3.12)$$

It is advantageous to rewrite the frequency domain expression for the scattered field in this form. It must be emphasized the the expression for $\mathbf{E}_{\text{scat}}^t$ in Equation (3.5) and the expression for $\mathbf{E}_{\text{scat}}^t$ in Equation (3.4) yield the same value, they are just different methods of expressing the same quantity. Also, the values for $\mathbf{E}_{\text{scat}}^t$ in both of those equations only differ from the value for \mathbf{E}_{scat} in Equation (2.18) by an $e^{j\omega t}$ term. The factor of 2 multiplying \vec{R}_{ot} accounts for the two-way travel of the electromagnetic fields (radar to target and target to radar). Note that now the scattered field has been expressed as the sum of the scattered fields from each of the Γ scattering centers.

This representation is useful for developing an expression for Doppler shift and also for developing an expression for angular error, as will be shown in Section 3.3. Next, bringing the $e^{j\omega t}$ term outside of the summation we have

$$\mathbf{E}_{\text{scat}}^t = e^{j\omega t} \sum_{i=1}^{\Gamma} E_i e^{j(\phi_i - 2k_o \hat{n}_i \cdot \vec{R})}. \quad (3.13)$$

Consider the phase angle of $\mathbf{E}_{\text{scat}}^t$ and define

$$\Phi = \angle \mathbf{E}_{\text{scat}}^t - \omega t = \phi - \omega t = \angle \mathbf{E}_{\text{scat}} \quad (3.14)$$

where ϕ was defined in Equation (2.23). Thus, Φ is the phase of the complex number \mathbf{E}_{scat} . The Doppler shift, f_d , is now defined as

$$f_d = \left(\frac{-1}{2\pi} \right) \frac{d\Phi}{dt}. \quad (3.15)$$

When there is no target-radar motion, the time derivative of $\frac{d\Phi}{dt}$ is zero and so is f_d . However, when there is radar-target motion, $\frac{d\Phi}{dt}$ is not equal to zero and we have a Doppler shift.

To calculate $\frac{d\Phi}{dt}$, let

$$\mathbf{E}_{\text{scat}}^t = (E_o e^{j\Phi}) e^{j\omega t} \quad (3.16)$$

where E_o (defined in Equation (2.23)) and Φ are real numbers and are the complex magnitude and phase, respectively, of $\mathbf{E}_{\text{scat}}^t$ with the $e^{j\omega t}$ dependence removed. Next, let

$$E_o e^{j\Phi} = G + jF \quad (3.17)$$

where G and F are both real numbers. Further, F and G can be written as

$$\begin{aligned} G &= \text{REAL} \{ E_o e^{j\Phi} \} = \sum_{i=1}^{\Gamma} E_i \cos(\phi_i - 2k_o \hat{n}_i \cdot \vec{R}_{ot}) \\ F &= \text{IMAG} \{ E_o e^{j\Phi} \} = \sum_{i=1}^{\Gamma} E_i \sin(\phi_i - 2k_o \hat{n}_i \cdot \vec{R}_{ot}) \end{aligned} \quad (3.18)$$

where the REAL operator takes the real part of a complex number and the IMAG operator takes the imaginary part. Then

$$\Phi = \arctan\left(\frac{F}{G}\right). \quad (3.19)$$

Taking the time derivative of Φ

$$\frac{d\Phi}{dt} = \frac{d}{dt} \left\{ \arctan\left(\frac{F}{G}\right) \right\} = \frac{1}{1 + \left(\frac{F}{G}\right)^2} \frac{GF' - FG'}{G^2} \quad (3.20)$$

where F' is the time derivative of F and G' is the time derivative of G . Rearranging terms and realizing that $F^2 + G^2 = E_o^2$

$$\frac{d\Phi}{dt} = \frac{1}{E_o^2} (GF' - FG'). \quad (3.21)$$

In the most general case, assuming that ϕ_i , \widehat{n}_i , and \vec{R}_{ot} are all functions of time, the time derivative of Φ reduces to

$$\frac{d\Phi}{dt} = \frac{1}{E_o^2} \sum_{i=1}^{\Gamma} \alpha_i \left\{ \frac{d\phi_i}{dt} - 2k_o \left(\frac{d\widehat{n}_i}{dt} \cdot \vec{R}_{ot} + \frac{d\vec{R}_{ot}}{dt} \cdot \widehat{n}_i \right) \right\} \quad (3.22)$$

where

$$\alpha_i = E_i \sum_{l=1}^{\Gamma} \cos\left(\phi_i - 2k_o (\widehat{n}_i \cdot \vec{R}_{ot}) - \phi_l + 2k_o (\widehat{n}_l \cdot \vec{R}_{ot})\right). \quad (3.23)$$

Finally, using Equations (3.15) and (3.22), an expression for the Doppler shift is

$$f_d = \frac{-1}{E_o^2} \sum_{i=1}^{\Gamma} \alpha_i \left\{ \frac{d\phi_i}{dt} - 2k_o \left(\frac{d\widehat{n}_i}{dt} \cdot \vec{R}_{ot} + \frac{d\vec{R}_{ot}}{dt} \cdot \widehat{n}_i \right) \right\}. \quad (3.24)$$

The Doppler shift is calculated by Equation (3.24), where α_i is defined in Equation (3.23), in the MATLAB code `scattered_field.m`. The three required time derivatives, $\frac{d\phi_i}{dt}$, $\frac{d\widehat{n}_i}{dt}$, and $\frac{d\vec{R}_{ot}}{dt}$, are calculated by taking the difference between their present values and their values at the previous iteration, and dividing by the time step, dt . This is the most accurate method available for calculating the Doppler shift. The example in Section 4 shows how the Doppler shift varies over a flight profile. The

range over which the Doppler shift varies as a function of time determines the required “Doppler bandwidth” of a radar (or missile) receiver. Also, using Equation (3.3), the frequency of the scattered field received by the radar, f_r , can be determined.

3.3 Angular Error

The angular error, or glint, is another important quantity for a radar or missile receiver. The angular error, or glint, is defined as the angular difference between the vector from the target phase center to the radar receiver and the direction of arrival of the electric field at the radar receiver. The direction of arrival (DOA) is defined as the direction of the vector normal to the incoming wavefront of the scattered electric field [15, 16, 17]. The DOA can also be defined as the direction of the Poynting Vector of the Transverse Electromagnetic Wave (TEM) incident at the radar receiver [17]. First, an expression for the DOA vector will be found. The DOA vector is a vector pointing in the DOA, and is called \vec{d} .

The DOA vector, \vec{d} , is related to the gradient of the phase of the scattered electric field incident at the radar receiver. Equation (3.19) gives an expression for the required phase, Φ . Rewriting this expression using the definitions of F and G , we obtain

$$\Phi = \arctan \left(\frac{\sum_{l=1}^{\Gamma} E_l \sin(\phi_l - 2k_o \hat{n}_l \cdot \vec{R}_{ot})}{\sum_{i=1}^{\Gamma} E_i \cos(\phi_i - 2k_o \hat{n}_i \cdot \vec{R}_{ot})} \right). \quad (3.25)$$

Using Equations (3.5) and (3.25), the DOA vector \vec{d} is related to the phase Φ by the gradient function, namely,

$$\vec{d} = -\nabla \Phi \equiv \hat{x} \frac{\partial \Phi}{\partial x} + \hat{y} \frac{\partial \Phi}{\partial y} + \hat{z} \frac{\partial \Phi}{\partial z} \quad (3.26)$$

where ∇ is the gradient operator and \hat{x} , \hat{y} , and \hat{z} are the unit vectors in the x , y , and z spatial Cartesian directions, respectively. The partial derivatives of Φ with respect to the x , y , and z -directions are easily computed. For a detailed computation of the partial derivatives, see [16]. Taking all three partial derivatives and combining the

results, a final expression for the DOA vector is

$$\vec{d} = -\nabla\Phi = \frac{2k_o}{E_o^2} \left\{ \sum_{i=1}^{\Gamma} \alpha_i \hat{n}_i \right\} \quad (3.27)$$

where α_i is defined in Equation (3.23) and \hat{n}_i is defined in Equation (3.12). The factor of 2 in front of k_o is due to the two-way propagation of the electric field from radar to target.

Next, recall the radar-target direction vector, \vec{R}_{ot} , defined in Equation (3.10), as the vector from the radar receiver to the phase center of the target. The angular error, Ω_e , is defined as the angular difference between the vectors \vec{d} and \vec{R}_{ot} . This angular error is calculated using the dot product as

$$\Omega_e = \arccos \left(\frac{\vec{d} \cdot \vec{R}_{ot}}{|\vec{d}| |\vec{R}_{ot}|} \right) \quad (3.28)$$

where \cdot denotes the vector dot product and $|\cdot|$ denotes the magnitude of the vector. The sign of Ω_e is determined using the cross product between \vec{d} and \vec{R}_{ot} .

The angular error is an important quantity since radar receivers and missile seekers point in the direction defined by the gradient of the incident field, the direction of \vec{d} , not in the direction of the target, \vec{R}_{ot} . Note that Ω_e is typically a very small number. In the simulations, Ω_e is calculated in the MATLAB code `scattered_field.m`. In the simulations, Ω_e is compared with the apparent angular wingspan of the C-29 aircraft as seen at the radar receive antenna.

4. Simulation Example

In this section, a typical simulation was performed and the output quantities are shown. First, the profile is described. Figure 4.1 shows a MATLAB plot of the simulated flight path of the C-29 aircraft. The radar (or missile seeker) is located at

Table 4.1: Parameters for sample simulation.

Variable	Description	Value
f	radar frequency	8.6674 GHz
P_{tr}	transmitter power	100 W
G_{tr}	transmitter antenna gain	1000
V_t	target velocity	150 m/s
\widehat{D}_t	target direction vector (x,y,z)	(1,0,0)
\vec{R}_{t_i}	init. target position (x,y,z)	(2800,-100,-100) m
\vec{R}_o	radar position (x,y,z)	(0,0,0) m
dt	time step for simulations	0.01 s
ti	number of iterations	601
t	time	0 to 6 s
n	normalized time	0 to 601

the origin of the coordinate system. Table 4.1 list the simulation parameters. Note that the values shown correspond to a full-scale C-29 aircraft. The aircraft had both rigid-body motion and individual scattering center motion. For the translational rigid-body motion, the standard deviation of each of the translational components was set to 10 cm, the cut-off frequency to 2 Hz, and the correlation matrix is as given in Equation (2.33). For the rotational rigid-body motion, the standard deviation of each of the angular components was set to 2° , the cut-off frequency to 2 Hz, and the correlation matrix is as given in Equation (2.33). The individual scattering center motion had a magnitude of 2 cm in each direction, with a cut-off frequency of 10 Hz, and the down-range and cross-range motion components are correlated by the matrix in Equation (2.38). These values were chosen in consultation with an aeronautical engineer [14].

The results of the simulation are shown in Figures 4.2 through 4.6. Examining

Figure 4.2, we can see the effects of the C-29 vibrations in the azimuth and elevation plots. The magnitude of the angular vibrations is smaller than the $\sigma_{rr} = 2^\circ$ due to the low pass filtering described in Section 2.5.1. Since the radar-target range is much greater than the vibrations, the effects of the vibrations can not be seen in the radar-target range plot. Also note C-29 Model Number 1, as defined in Table 2.3, was the best model to use for this simulation, and was used exclusively throughout this simulation run. Recall that the center AZ and EL values for Model Number 1 are -3.76° and 3° , respectively. These values are closer to the required AZ and EL values shown in Figure 4.2 than the values for any other available C-29 models listed in Table 2.3.

Figure 4.3 shows both the scattered electric field at the radar receiver, $E_s(t)$ as defined in Equation (2.24), plotted in dB, and the angular error at the radar receiver, Ω_e as defined in Equation (3.28), plotted in degrees, both versus time. The rapid variations in the scattered field can mainly be attributed to the $\cos(\omega t)$ dependence of $E_s(t)$. The other variations in $E_s(t)$ can be attributed to the dynamics of the scenario, along with the rapidly changing scattering characteristics of the C-29 as a function of AZ and EL angles. The angular error, Ω_e , shown in Figure 4.3, is referenced against the angular extent (or wingspan) of the C-29 aircraft as seen at the radar receiver. From the plot it can be seen that the angular error sometimes exceeds the wingspan of the aircraft, but usually is smaller than the wingspan. The angular error does rapidly change, as expected [14]. In order to see more detail of the scattered field and the angular error, “blow-ups” of the two plots are shown in Figure 4.4. In this figure, the quantities are shown for time $t = 0$ to 2 seconds.

Figure 4.5 shows the Doppler frequency, f_d as defined in Equation (3.24), and the received power at the radar, Pr as defined in Equation (3.1), both versus time. As with the scattered field, “blow-up” plots are shown in Figure 4.6. A radar receiver must be able to track the changing frequency as a function of time. Recall, the CW

frequency received by a radar, f_r , is equal to the Doppler frequency, f_d , plus the transmitter frequency, f . Thus, $f_r = f + f_d$. The Doppler bandwidth is a term which refers to the range of frequencies over which f_r ranges. Since f does not change with time, the Doppler bandwidth is determined by the Doppler shift, f_d . This simulation shows that the Doppler bandwidth is relatively large, with a range of 10 kHz. Examining the "blow-up" plot in Figure 4.6, the Doppler bandwidth is in the range of 300 to 3000 Hz for times durations up to 0.1 seconds. Obviously, f_d is a function of time, and over a long time, f_d may change by many kilohertz, but for short time durations, under 0.1 seconds, the change in f_d does not exceed 2000 Hz. At the times when f_d has large jumps, this usually can be attributed to low scattered field values. This is verified by comparing Figures 4.3 and 4.5. The Doppler bandwidths observed here seem to be rather larger for the scenario, which might suggest more experimentation with the rigid-body and individual scattering center motion model parameters (magnitude, cut-off frequency, and correlation matrices).

Obviously, only one simulation example has been shown here. As the dynamic situation changes, different phenomena may arise. Also, turning off or changing the various aircraft vibration parameters will significantly effect the scattered field, angular error, and Doppler frequency.

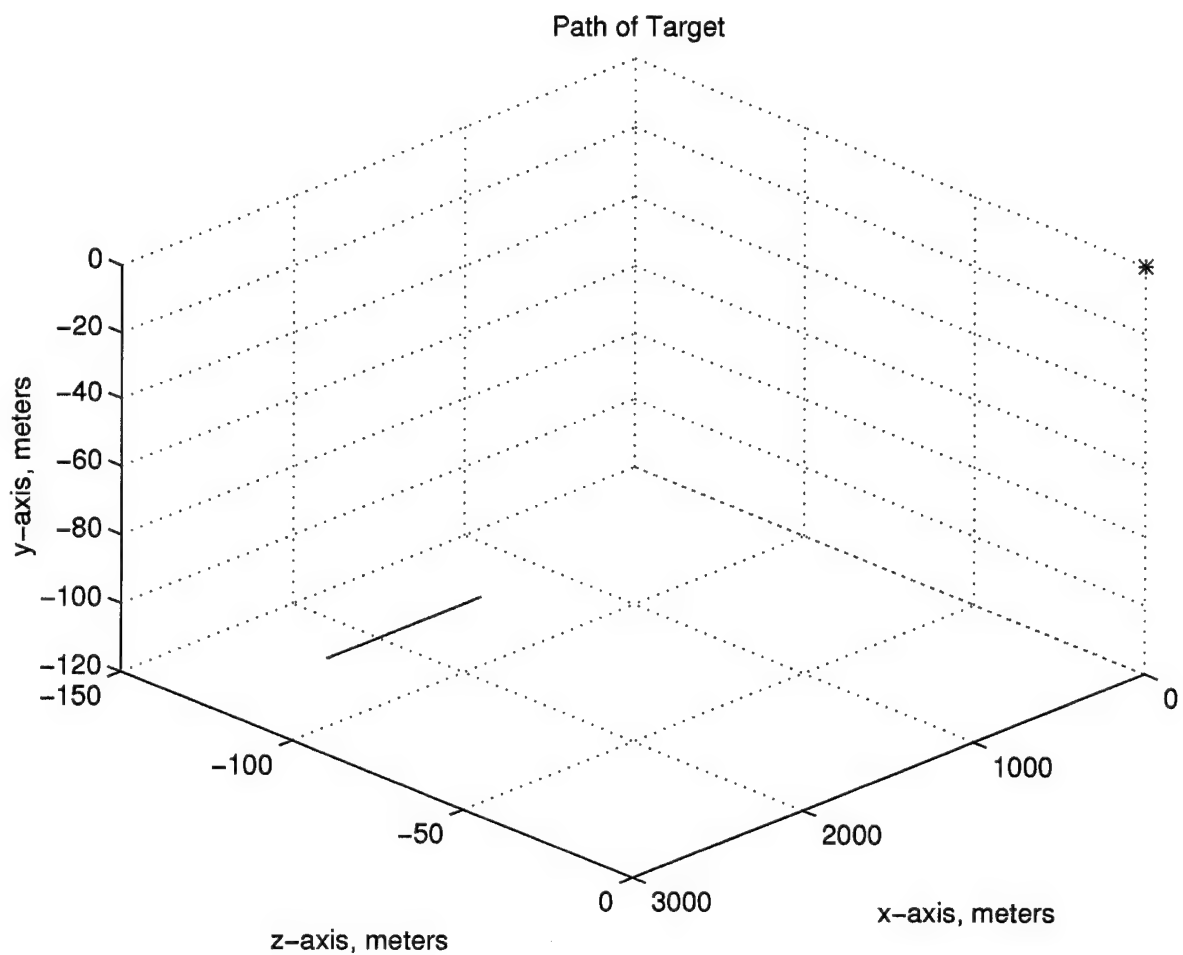


Figure 4.1: Path of C-29 aircraft for sample simulation in (x, y, z) space. The C-29 is flying straight and level, with vibrations as described in the text, along the line shown in the plot. The radar is located at the origin, denoted by a '*'.

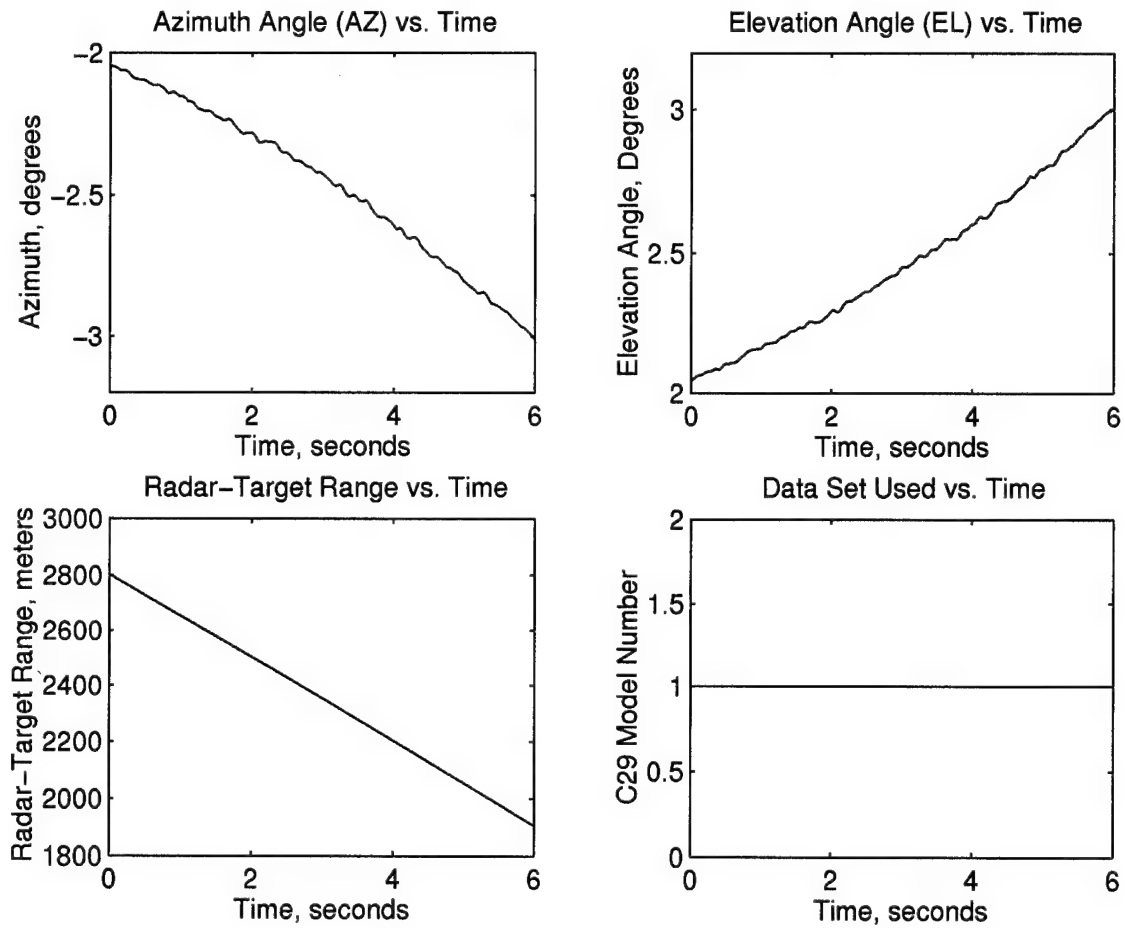


Figure 4.2: Azimuth and elevation angles for sample simulation, as defined in Figure 2.2, versus time, relative to the aircraft pointing vector $\hat{P}t$. Also, radar-target range versus time and C-29 Model Number used versus time.

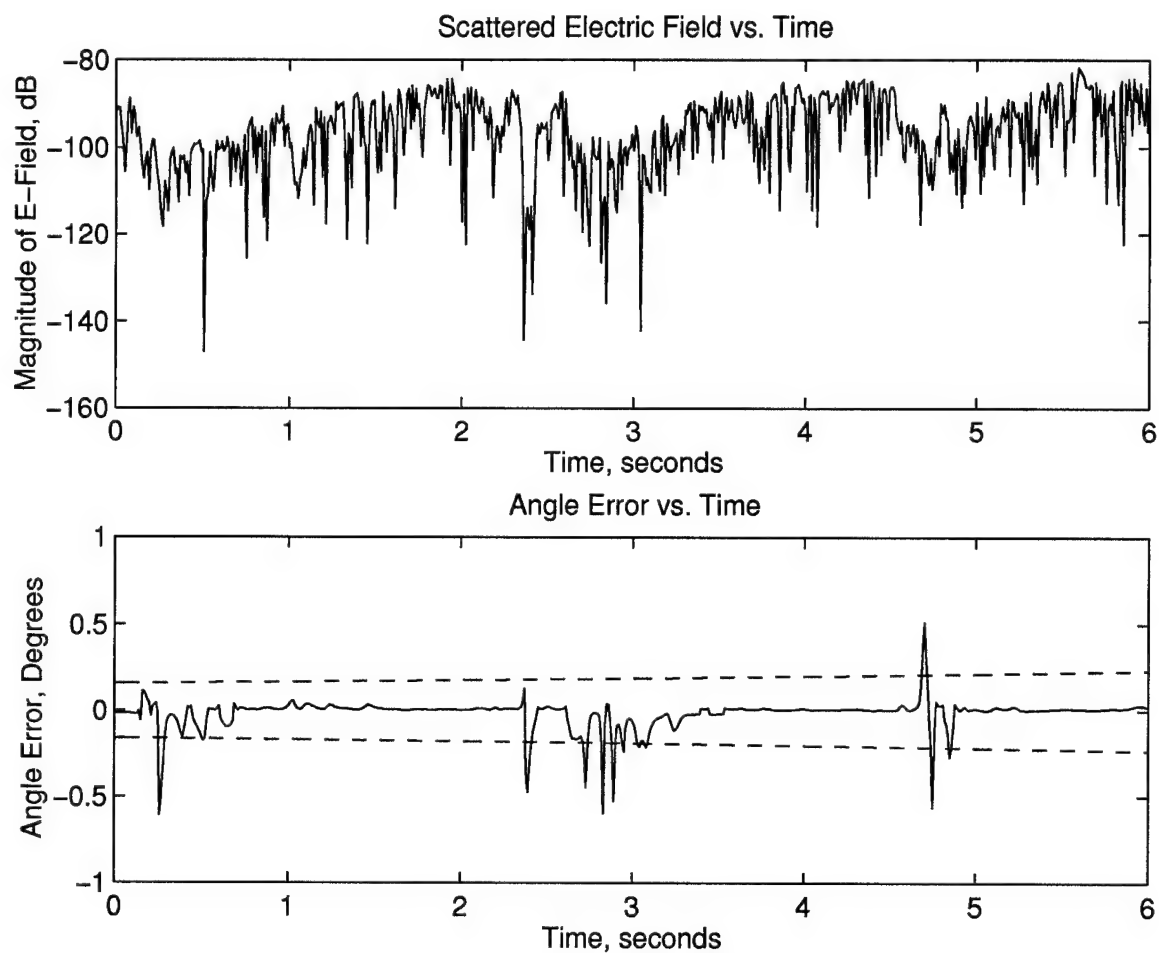


Figure 4.3: Scattered Electric Field and Angular Error versus time for sample simulation. The dashed lines in the angular error plot denote the relative angular wingspan of the aircraft as seen by the radar receiver.

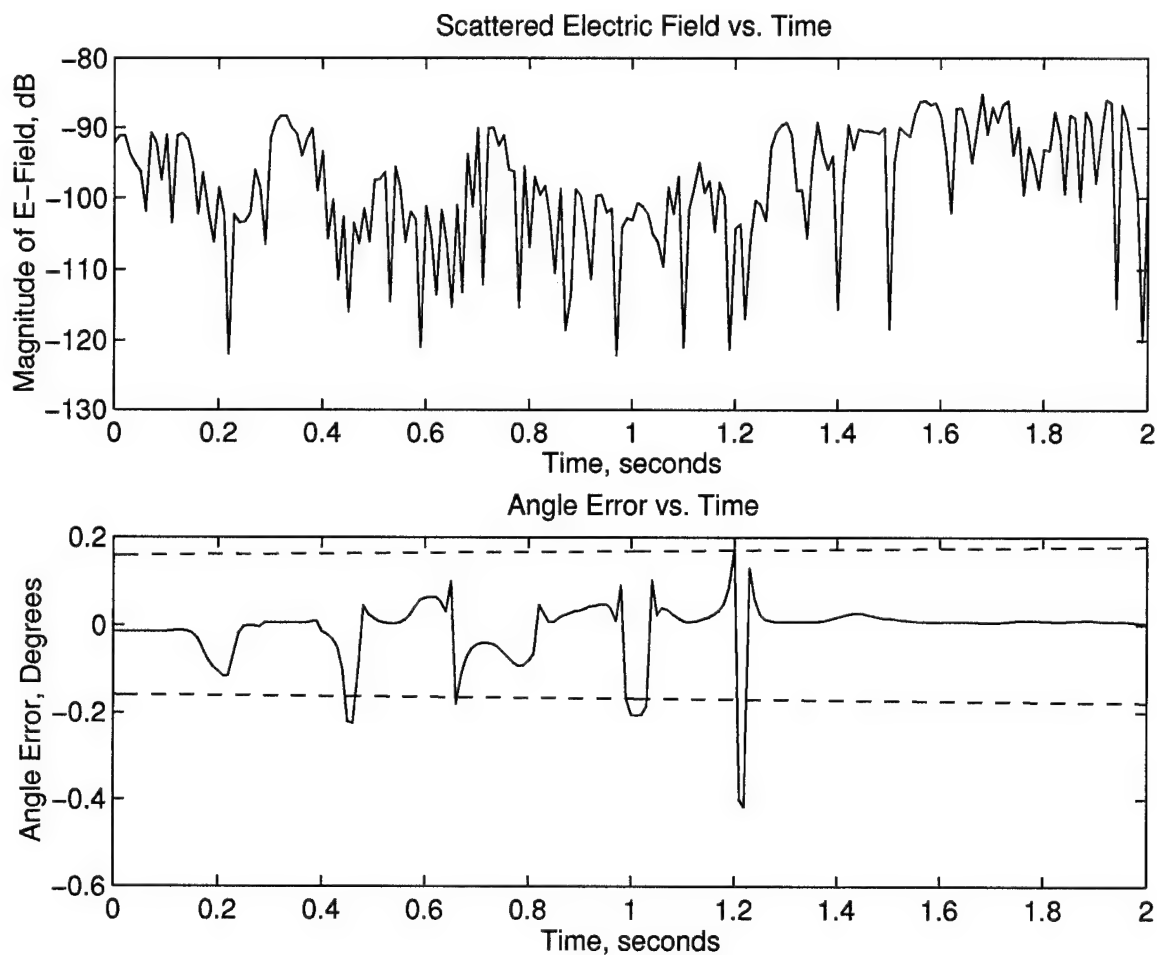


Figure 4.4: Blow-up of Scattered Electric Field and Angular Error versus time for a portion of the sample simulation. The dashed lines in the angular error plot denote the relative angular wingspan of the aircraft as seen by the radar receiver.

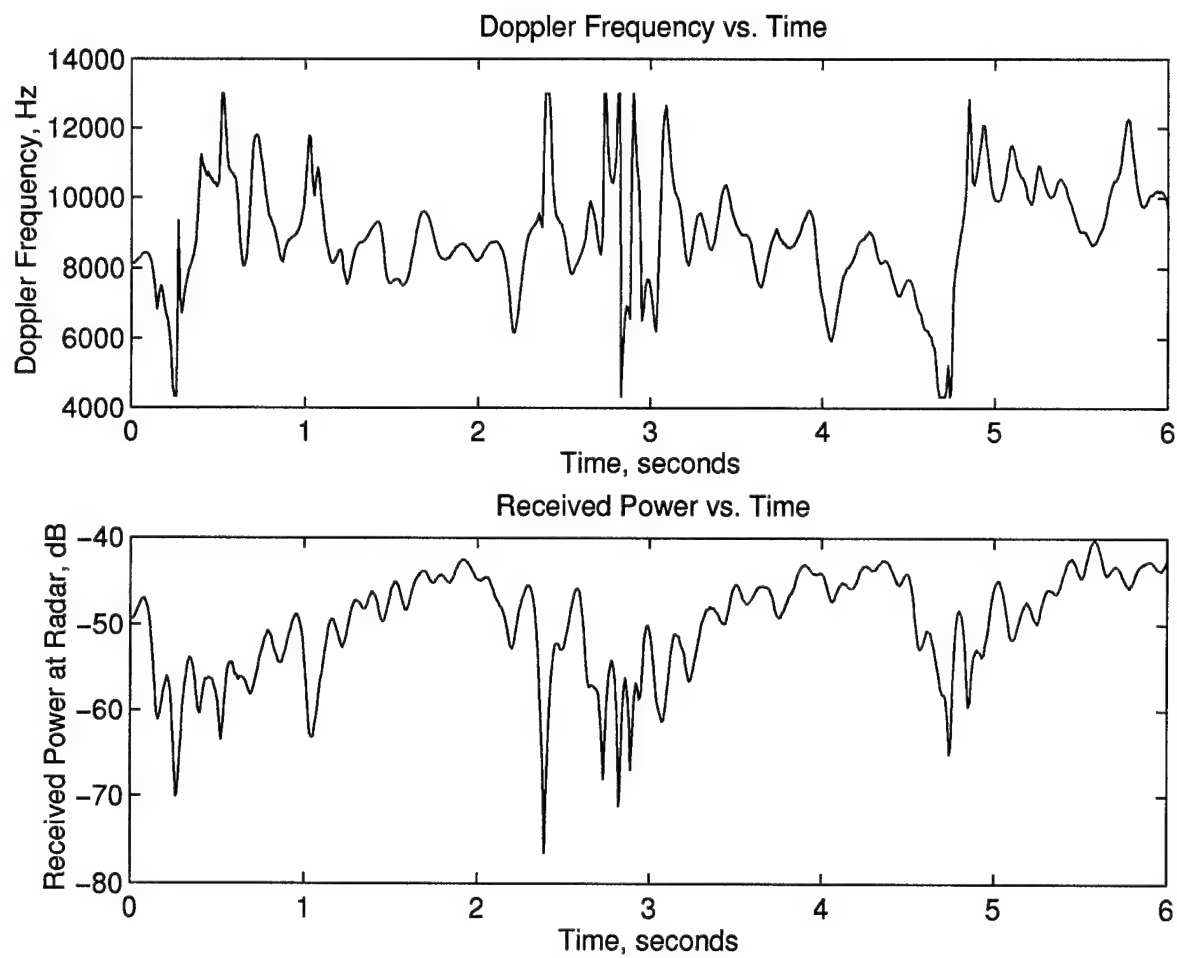


Figure 4.5: Doppler frequency and Received Power versus time for sample simulation.

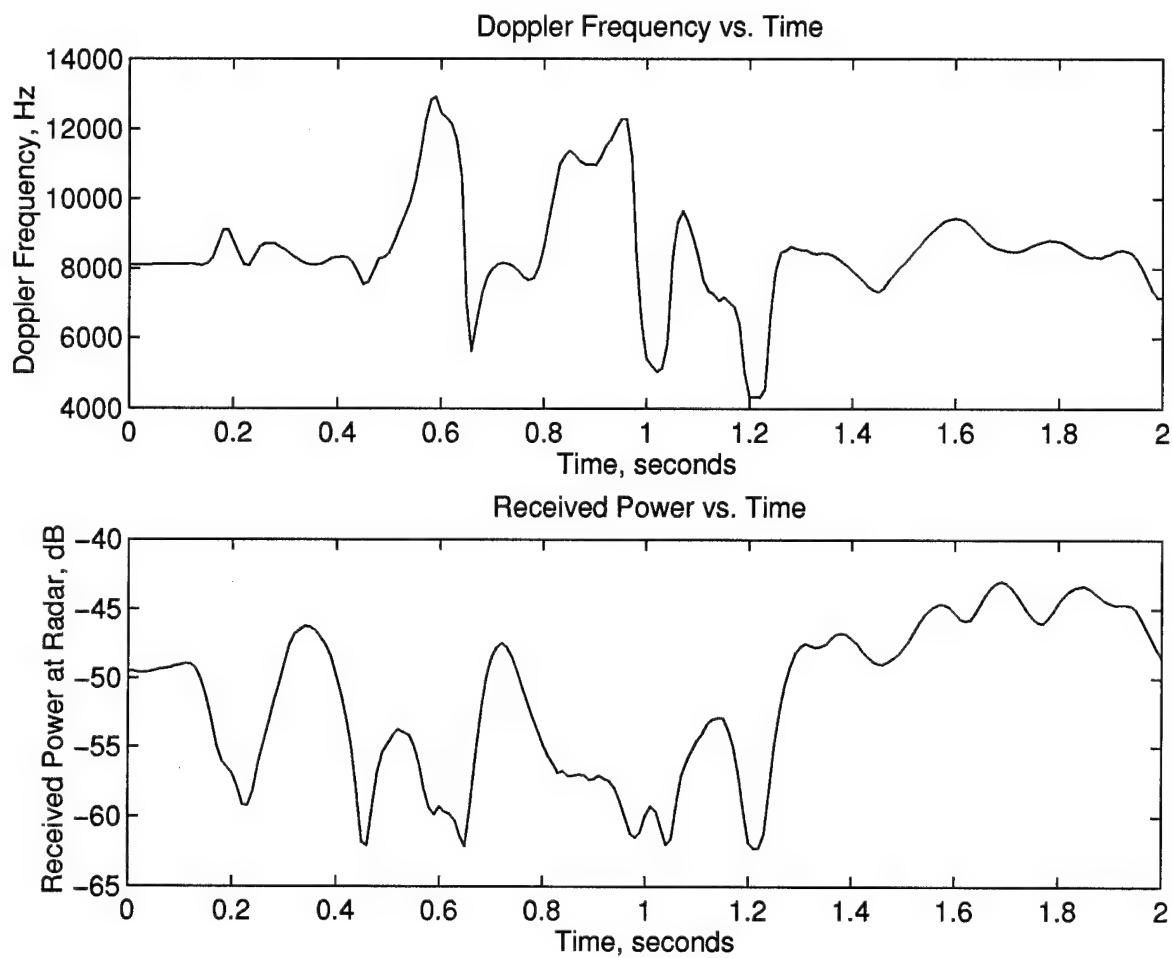


Figure 4.6: Blow-up of Doppler frequency and Received Power versus time for a portion of the sample simulation.

5. MATLAB Code Descriptions

In this section, all of the MATLAB codes used in the simulation package are briefly described. Some of the codes are listed in Appendix B. Table 5.1 lists all of the MATLAB codes needed to run the simulations. In Table 5.1, the code's name, a brief description of the code, and the type of MATLAB code is specified for each code. The two type of MATLAB codes are script and function. A script code can be run from the MATLAB command line or as a background job. A function code needs to be properly called by a script function or from the command line, and is analogous to a subroutine. For more information concerning MATLAB, consult [18]. The simulations were performed using MATLAB version 4.2c, dated December 31, 1994, SUN UNIX. MATLAB also has versions which run on IBM PCs and APPLE machines.

As stated before, to run a dynamic simulation, the only code that needs to be executed in MATLAB is **profile.m**. This script file calls all of the other required codes. Also, all options are definable in **profile.m**, which passes the required information to

Table 5.1: Listing of MATLAB codes.

Code Name	Description	Type
profile.m	runs entire dynamic simulation	script
scattered_field.m	calcs various outputs	function
rigid_motion.m	calcs rigid body motion	function
indiv_mot.m	calcs individual motion	function
model_image.m	plots a model ISAR image	script
C29model.m	holds all C29 models	function
makemodel.m	makes C29 models	script
prony2d.m	runs 2-D TLS Prony Technique	function
C29image.m	makes C-29 ISAR image	script

the other MATLAB codes. The script file **profile.m** is commented and the variable names have been chosen to minimize confusion and to correlate with the equations in this report as closely as possible.

The CPU time of the simulation performed in Section 4 is on the order of two minutes being run on a SUN SPARC 2 with 32 MB of RAM. Little effort was given to speed-up the simulations, but since many loops are in the MATLAB code, if the simulations were rewritten in a faster language, such as "C", then the required CPU time should decrease significantly. MATLAB is an "interpreter" language, and is not efficient at performing loops.

There should be no problem converting the MATLAB codes to another language if "real-time" operation is required. Note that it will only be necessary to convert the following codes from MATLAB to the new language: **profile.m**, **scattered_field.m**, **rigid_motion.m**, **indiv_mot.m**, **C29model.m**, and **model_image.m**. The 2-D TLS-Prony Technique codes do not need to be converted as they are not used in the simulations, they only create the models. The only built-in MATLAB functions which are used in the simulations which are not "simple" functions easily performed in any language, are the filter design functions, **fir1.m**, used in **rigid_motion.m** and **indiv_motion.m**. The filter designs can be performed in MATLAB, and the results (filter coefficients) can be hard-coded into any new code.

6. Conclusions and Recommendations

The goal of this simulation package is to provide realistic scattered electric field, Doppler shift, and angular error values for a dynamic radar-target engagement. These values can be used in a hardware-in-the-loop simulation of a radar-target or missile-target encounter to help evaluate the performance of radar receivers or missile seekers.

It must be emphasized that a mono-static radar or fully-active missile seeker has been assumed as the radar in the scattering models. However, if bi-static scattering data were available, then the simulations can be corrected for this situation.

The hardware-in-the-loop simulation would have to simulate the scattered electric field, the Doppler shift, and the angular error values predicted by the simulations. Details of how this is accomplished are given in [16]. With the resources available to DSTO, this should easily be accomplished.

One phenomena which is not taken into account in the simulations presented here are Jet Engine Modulation (JEM) effects on the scattered field. JEM effects occur due to the turbine blades in the target aircraft's jet (or turbo-prop) engines rotating at high rates of speed. JEM effects will change the scattered field, angular error, and Doppler shift values. For nose-on scenarios (aircraft heading directly toward radar), the JEM effects will be most significant. For other radar-aircraft aspect angles, the JEM effects will be less significant. The JEM effects tend to modulate the scattered field, and possibly these effects can be modeled if the simulations are upgraded. Even though the simulations do not take JEM effects into account, the results are still valid and the JEM effects are secondary in most scenarios to the more important effects of aircraft motion (rigid body and individual).

Another limitation of the simulations is in the individual motion algorithm in Section 2.5.2. This algorithm moves the scattering centers around randomly about their unperturbed positions without any regard to the scattering centers location on the aircraft. As a possible upgrade to the simulations, some weighting factor as a function of scattering center location could be added to the motion values allowing scattering centers further away from the phase center of the target to move more than scattering centers which are near the phase center of the target. This is meant to simulate scattering centers on wing-tips moving more than scattering centers on the fuselage.

In the simulations it is assumed that the target (C-29 aircraft) is flying and vibrating, and the radar is stationary at position $\vec{R}o$. If the radar (or missile seeker) is also traveling at a velocity and vibrating, the simulations can be upgraded to account for this by varying $\vec{R}o$ and also accounting for bore-sight angle errors.

Taking the above concerns into account, the simulations presented contain a very accurate model, the 2-D Damped Exponential Model, whose parameters were derived directly from ISAR data, for the scattering from a radar target. The signatures shown in Section 4, when compared with actual dynamic signatures [19], show that there is an overall correspondence between the results of the simulations and what is observed in reality. Thus, it is concluded that this simulation package would present actual radar or missile receiver hardware with signatures that are realistic for a dynamic radar-target scenario.

7. References

- [1] J. J. Sacchini, A. Romano, and W. M. Steedly, "Evaluation of single and full-polarization two-dimensional prony techniques applied to radar data," in *1994 SPIE Proceedings, Orlando, FL*, vol. 2234, Apr. 1994. (to appear).
- [2] J. J. Sacchini, *Development of Two-Dimensional Parametric Radar Signal Modeling and Estimation Techniques with Application to Target Identification*. PhD thesis, The Ohio State University, Columbus, OH, 1992.
- [3] J. J. Sacchini, W. M. Steedly, and R. L. Moses, "Two-dimensional prony modeling and parameter estimation," *IEEE Transactions on Signal Processing*, vol. 41, pp. 3127 – 3137, Nov. 1993.
- [4] M. I. Skolnik, *Introduction to Radar Systems*. New York: McGraw-Hill, 1980.
- [5] N. F. Chamberlain, "Recognition and analysis of aircraft targets by radar, using structural pattern representations derived from polarimetric signatures," June 1989. Also: ElectroScience Laboratory Report 719710-3.
- [6] J. R. Huynen, *Phenomenological Theory of Radar Targets*. PhD thesis, Technical University Delft, Druckerij Bronder-Offset N.V. Rotterdam, Netherlands, 1970.
- [7] W. M. Boerner, M. B. El-Arini, C. J. Chen, and P. M. Mastorius, "Polarization dependence in electromanetic inverse problems," *IEEE Transactions on Antennas and Propagation*, vol. AP-29, pp. 262–271, Mar. 1981.
- [8] D. L. Mensa, *High Resolution Radar Cross-Section Imaging*. Norwood, MA: Artech House, 1 ed., 1991.
- [9] B. P. Anderson and J. J. Sacchini, "A comparison of stepped and continuous-wave radar imaging algorithms," in *Proceeding of the 1994 NAECON Conference*, (Dayton, OH), pp. 303 – 308, IEEE, May 1994.
- [10] D. L. Hardesty, "An investigation into the application of one and two dimensional parametric signal processing techniques to the extension of radar data," Master's thesis, Air Force Institute of Technology, 1993.
- [11] A. Romano, "Application of two-dimensional parametric signal processing techniques to the radar target identification problem," Master's thesis, Air Force Institute of Technology, 1993.
- [12] J. D. Pullis, "Three-dimensional inverse synthetic aperture radar imaging," Master's thesis, Air Force Institute of Technology, 1995.

- [13] A. V. Oppenheim and R. W. Schaffer, *Discrete-Time Signal Processing*. Englewood Cliffs, NJ: Prentice Hall, 1989.
- [14] Defence Science and Technology Organisation, Salisbury, South Australia, AUSTRALIA, Apr. 1996. Personal Communication with Dr. Mark Petrusma, Aeronautical Engineer.
- [15] J. D. Kraus and K. R. Carver, *Electromagnetics*. New-York, NY: McGraw-Hill, 1 ed., 1973.
- [16] R. Glavacich, "The adaptation of vector algebra to the analysis of rf echoes from multi-point targets," Tech. Rep. DSTO-TR-XX, Defence Science and Technology Organisation, Salisbury, South Australia, AUSTRALIA, May 1996.
- [17] R. F. Harrington, *Time-Harmonic Electromagnetic Fields*. New-York, NY: McGraw-Hill, 1961.
- [18] MathWorks, *Matlab User's Guide*. Natick, MA, 01760: The MathWorks, Inc., 1992.
- [19] Defence Science and Technology Organisation, Salisbury, South Australia, AUSTRALIA, Apr. 1996. Personal Communication with Dr. Noel Martin.
- [20] R. M. Lewitt, "Reconstruction algorithms: Transform methods," in *Proc. IEEE*, vol. 71, pp. 390-408, Mar. 1983.
- [21] R. L. Bracewell, *The Fourier Transform and Its Applications*. New York, NY: McGraw-Hill, 2 ed., 1986.

8. Appendices

Appendix A. Radar Imaging Using Wide-Bandwidth Data

This appendix briefly describes how an Inverse Synthetic Aperture Radar (ISAR) Image is formed when multiple frequency multiple angle ISAR data is available. The analysis of 2-D multiple frequency and multiple angle radar data has mainly utilized the 2-D Fourier Transform [20]. These techniques are also applied in tomography and have applications in electron microscopy, x-rays, radio astronomy, geology and medical imaging.

The data analyzed in this effort is ISAR data, however, Synthetic Aperture Radar (SAR) data could also be analyzed. The difference between a SAR scenario and an ISAR scenario is that, in a SAR scenario the target is stationary and the radar is moving while in an ISAR scenario, the radar is stationary and the target is moving. In either case, when a measurement is taken it is assumed that both radar and target are stationary. Any motion present during a measurement must be “subtracted” out [8]. Thus, it is assumed in this Appendix that the measurements are taken are done so for a static (no-motion) situation.

First, the relationship between the frequency domain and the time domain will be established. The 2-D image (SAR or ISAR) of a radar target is sometimes called “the reflectivity density function” [8] and is denoted $h(x, y)$ in this appendix.

The radar-target geometry for 2-D imaging purposes is shown in Figure 8.1. The uv -coordinates are fixed to the radar while the xy -coordinates are fixed to the target. The range between the maximum extent of the target and the radar is assumed to be much greater than the maximum extent of the target. Also, the target is assumed to be in the far field of the radar. When measurements are taken by the radar, it is assumed that there is no relative velocity between the radar and the target. As a start, it is assumed that the radar is radiating a single frequency, f . For a fixed ψ (ψ in this appendix and θ used in the main body of the report are equivalent angular variables) and a particular range v , the signal received by the radar is

$$p(v, \psi) \propto \int_{-\infty}^{\infty} h_{\psi}(u, v) du \quad (8.1)$$

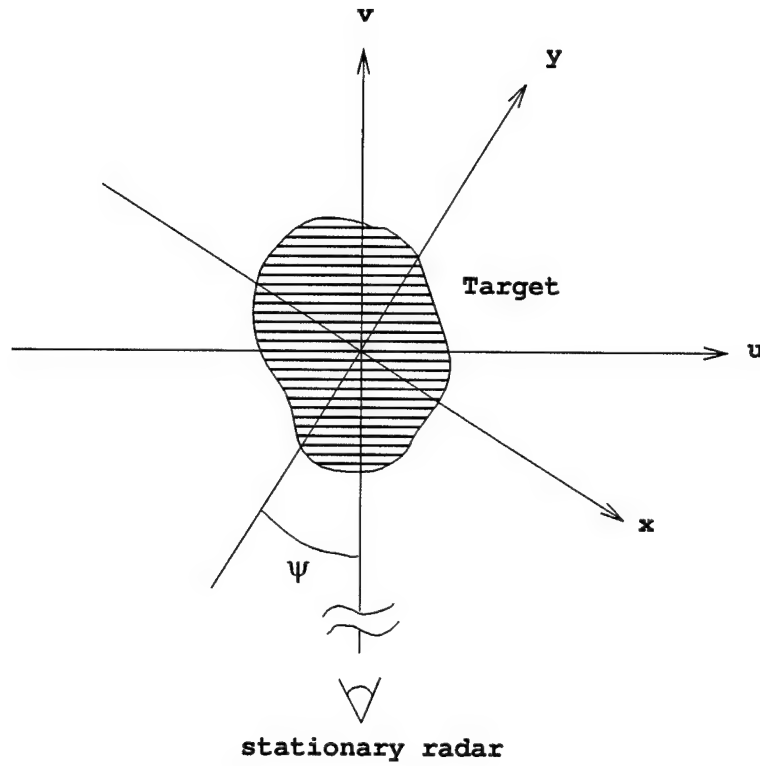


Figure 8.1: Radar-target geometry for 2-D imaging.

where

$$h_{\psi}(u, v) = h(x, y) \quad (8.2)$$

and

$$\begin{aligned} x &= u \cos(\psi) - v \sin(\psi) \\ y &= u \sin(\psi) + v \cos(\psi). \end{aligned}$$

The total signal received at a given ψ , $H(\psi)$, is an integral of all signals along the projection of $h(x, y)$ onto the v -axis modified by a phase factor which accounts for the round-trip phase delay along the v -axis. Thus,

$$H(\psi) = \int_{-\infty}^{\infty} p(v, \psi) e^{\frac{-j4\pi v}{\lambda}} dv. \quad (8.3)$$

Utilizing Equation (8.1), Equation (8.3) becomes

$$H(\psi) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h_{\psi}(u, v) e^{\frac{-j4\pi v}{\lambda}} du dv. \quad (8.4)$$

Realizing $v = y \cos(\psi) - x \sin(\psi)$, using Equation (8.2) and defining

$$f_x = \frac{2 \sin(\psi)}{\lambda} \quad (8.5)$$

$$f_y = \frac{-2 \cos(\psi)}{\lambda}, \quad (8.6)$$

the received signal at the radar becomes

$$H(f_x, f_y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h(x, y) e^{-j2\pi(f_x x + f_y y)} dx dy. \quad (8.7)$$

This is the standard 2-D Fourier Transform. All of the useful properties which exist for the Fourier Transform exist for the 2-D Fourier Transform pair

$$h(x, y) \Longleftrightarrow H(f_x, f_y). \quad (8.8)$$

From Fourier Theory [21],

$$h(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} H(f_x, f_y) e^{j2\pi(f_x x + f_y y)} df_x df_y. \quad (8.9)$$

This relationship can be written in polar form by defining $f = \sqrt{f_x^2 + f_y^2}$. Thus

$$h(x, y) = \int_0^{\infty} \int_0^{2\pi} H(f, \psi) e^{-j2\pi f(x \sin(\psi) - y \cos(\psi))} d\psi df \quad (8.10)$$

relates the reflectivity density function to the frequency domain data. Equations (8.1) through (8.10) form the Projection-Slice Theorem [20].

In practice, we will have samples of $H(f, \psi)$, as in Figure 8.3, and wish to approximate $h(x, y)$ from these samples. The samples of $H(f, \psi)$ are usually the scattering coefficients of the target as a function of frequency and angle ($S(f, \theta)$). Usually one angular dimension is fixed and the other is varied to yield a 2-D "slice" of the target. However, any angular slice of the target can be used to generate an image.

There are several methods which are used to approximate $h(x, y)$ from the samples of $H(f, \psi)$. If the samples of $H(f, \psi)$ are on a polar grid, as in Figure 8.3, one method

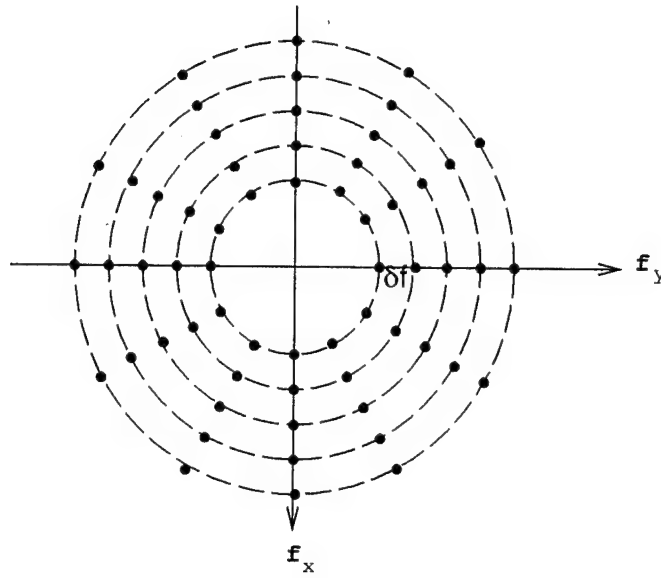


Figure 8.2: 2-D frequency domain data set for full angular coverage.

is to interpolate the data to a rectangular grid and apply the 2-D Inverse Fast Fourier Transform (2-D IFFT). Also, if the polar grid is “almost” rectangular, no interpolation is performed and it is assumed the polar grid data lies on a rectangular grid. The 2-D IFFT is directly applied to the polar grid data. This is the case for all the images appearing in this report (Figures 2.5, 2.6, 2.7, and 2.8). Another method is to numerically approximate the double integral in Equation (8.10). Yet another option is to use the convolution-backprojection method [20]. One type of data set in the frequency domain is shown in Figure 8.2, which corresponds to full angular coverage. Figure 8.3 shows the case of the data available for limited angular coverage.

In practice, data is usually only available over a limited angular extent (usually the maximum angular extent is between 3° and 20° for spotlight mode SAR [8]). For the data associated with the C-29 scale model, the angular extent is 10° and the angular swath is in the AZ direction.

For the SAR images in Figures 2.5, 2.6, 2.7, and 2.8, the radar image was formed using the 2-D IFFT with no interpolation. The de-focusing effects of the polar grid data were not degrading enough to require focusing (interpolation of 2-D frequency domain data from polar grid to rectangular grid). Also, the 2-D Damped Exponential

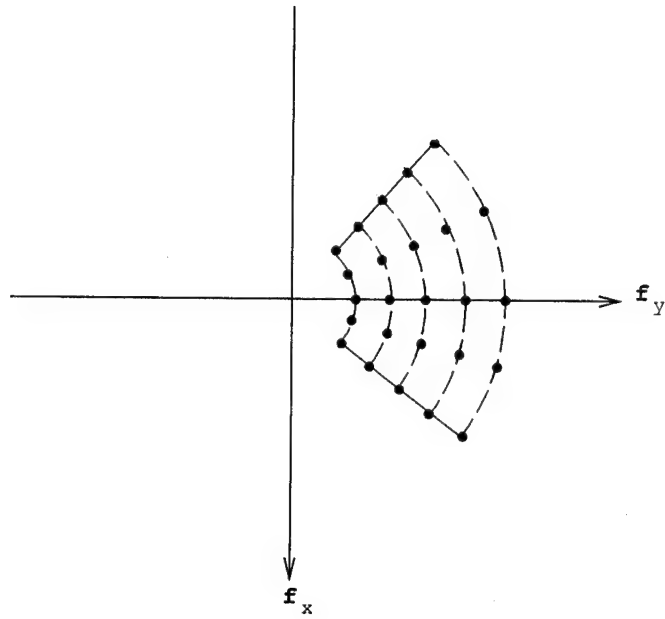


Figure 8.3: 2-D frequency domain data set for limited angular coverage.

parameters, $\{\mathbf{p}_{\mathbf{x}_\gamma}, \mathbf{p}_{\mathbf{y}_\gamma}, \mathbf{a}_\gamma\}_{\gamma=1}^\Gamma$, were estimated directly from the un-focused data set, defined by D in Equation (2.7).

Appendix B. MATLAB Code Listings

The following codes are listed in this Appendix: `profile.m`, `scattered_field.m`, `rigid_motion.m`, `indiv_mot.m`, and a portion of `C29model.m`. Descriptions of these codes are given in Section 5.

```
%      %%%%%%%%%%%%% profile.m %%%%%%%%%%%%%

% MATLAB script file profile.m
% Written by J. Sacchini, DSTO, Adelaide, 1 May 96
% See final report for explanation
%
% Run a Typical profile
%
% All angles in degrees, all frequencies in hertz, all distances in
% meters, powers in watts, gains in numeric (not dB)
% To turn on a particular feature, make the *_on variable = 1, to turn
% off set it to 0 (actually, anything not equal to 1).

clear %Initially clear out all variables

% Initially produce SAR image, if desired, must guess at AZ and EL angles
iimage=0;
if iimage==1,
    f=26.0023*10^9/3;
    AZim=-3;
    ELim=3;
    [px,py,a,azmid,el,fmin,df,nf,da,na,choice]=C29model(AZim,ELim,f);
% [px,py,a,azmid,el,fmin,df,nf,da,na,model_no]=point_model(AZdeg,ELdeg,f);
    model_image(px,py,a,azmid,el,fmin,df,nf,da,na,f);
    hold on
end

% Input desired frequency, transmitter power and antenna gain, NOT in dB
f=26.0023*10^9/3;
Ptr=100;
Gtr=1000;

% Do you wish warnings of using models outside of their intended
% angular areas and do you wish the scattering centers to be plotted
warnings=1;
```

```

scplot=0;

% Input radar position (fixed for now)
Ro=[0 0 0];

% Start profile, ti=initial time, tv=time vector,
% move target, keep observation point fixed (for now)
ti=0;
dt=0.01;
tv=[0:dt:6];
%tv=0;
tv=tv(:);
iterations=length(tv);

% Calculate flight path positions, without any aircraft translational
% or rotational shaking
Vt=150; % velocity of target
Dt=[-1 0 0]; % direction of target
Dt=Dt/norm(Dt); % normalize direction
Rtinit=[2800 -100 -100]; %initial target position
Rtv_nos=(Rtinit'*ones(1,iterations))'+Vt*(Dt'*tv)';
Dtv_nos=(Dt'*ones(1,iterations))';

% Put in fixed roll, pitch, and yaw angles if desired
Roll_Angle=0; % Constant Roll Angle
Pitch_Angle=0; % Angle of Attack
Yaw_Angle=0; % Crab Angle
RPYv_nos=[Roll_Angle*ones(iterations,1) Pitch_Angle*...
ones(iterations,1) Yaw_Angle*ones(iterations,1)];

% Translational and rotational motion shaking (rigid aircraft)
randn('seed',1) % start with same seed each time if desired
trans_on=1;
fc_trans=2; % Cut-off freq of translational shaking in Hz
mag_trans=0.1; % Standard Deviation of Magnitude of shaking in meters
rot_on=1;
fc_rot=2; % Cut-off freq of rotational shaking in Hz
mag_rot_deg=2; % Standard Deviation of Angle shaking in degrees
[Rtv_s,Dtv_s,RPYv_s]=rigid_motion(Rtv_nos,Dtv_nos,RPYv_nos,tv,...
dt,trans_on,fc_trans,mag_trans,rot_on,fc_rot,mag_rot_deg);
Rtv=Rtv_s; %position

```

```

Dtv=Dtv_s; %direction of flight, not implemented
RPYv=RPYv_s; %roll,pitch,yaw in degrees

%%%%%%%%% Insert other rigid body motion models here - Dr. Mark Petrusma
% make Rtv be the x,y,z perturbations in meters, Rtv has
% iterations rows and 3 columns
% Make RPYv be the roll, pitch, yaw perturbations in degrees, RPYv has
% iterations rows and 3 columns
% Make sure if you use alternate models, to turn off trans and rot
% motion above

% Individual Scattering Center Motion (Flexing Aircraft)
indiv_mot_on=1;
fc_indiv=10;
mag_indiv_mot=0.02;
nkeep=10; % must specify how many scattering centers to move independently
IMvx=zeros(iterations,nkeep);
IMvy=zeros(iterations,nkeep);
for k=1:nkeep,
    IMvx(:,k)=indiv_mot(tv,dt,indiv_mot_on,fc_indiv,mag_indiv_mot);
    IMvy(:,k)=indiv_mot(tv,dt,indiv_mot_on,fc_indiv,mag_indiv_mot);
end

% Now run profile for all times in tv and calculate all quantities
%
for ti=1:iterations,
    t=tv(ti);
    Rt=Rtv(ti,:);
    Rt_nos=Rtv_nos(ti,:);
    Dt=Dtv(ti,:);
    RPY=RPYv(ti,:);
    if ti==1, Rt_delay=Rtv(1,:); vt_old=0; Rsc_old=0; end
    if ti~=1, Rt_delay=Rtv(ti-1,:); end
% Dt=-(Rt-Ro); % to point directly at obs point
    aRto(ti)=norm(Rt-Ro); % Range between target and observation point
    [Es(ti),Es_complex(ti),DOA(ti),Pr(ti),fd(ti),AZdeg(ti),ELdeg(ti),...
    azmid(ti),el(ti),model_no(ti),vt_old,Rsc_old,pha_old,extent_deg(ti)...
    ,dvec(ti,:),d(ti,:)] = scattered_field(t,f,Ro,Rt,Rt_nos,vt_old,...
    Rsc_old,pha_old,dt,Dt,RPY,Vt,Ptr,Gtr,warnings,scplot,ti,IMvx,...
    IMvy,indiv_mot_on);
end

```

```

if iterations>1, fd(1)=fd(2); end
% Look for places where we changed model numbers, and throw out
% doppler because it is incorrect during model changeovers.
model_diff=model_no-[model_no(1) model_no(1:iterations-1)];
ich=find(model_diff~=0);
fd(ich)=fd(ich-1);

% Plot output data, if desired
iplot=1;
if iplot==1,
    figure
    subplot(2,1,1)
    plot(tv,fd)
    xlabel('Time, seconds')
    ylabel('Doppler Frequency, Hz')
    title('Doppler Frequency vs. Time')
    subplot(2,1,2)
    plot(tv,10*log10(Pr))
    xlabel('Time, seconds')
    ylabel('Received Power at Radar, dB')
    title('Received Power vs. Time')
    figure
    subplot(2,1,1)
    plot(tv,20*log10(abs(Es)))
    xlabel('Time, seconds')
    ylabel('Magnitude of E-Field, dB')
    title('Scattered Electric Field at Receiver vs. Time')
    subplot(2,1,2)
    plot(tv,DOA)
    xlabel('Time, seconds')
    ylabel('Angle Error, Degrees')
    title('Angle Error vs. Time')
    hold on
    plot(tv,extent_deg/2,'--')
    plot(tv,-extent_deg/2,'--')
    hold off
    figure
    subplot(2,2,1)
    plot(tv,AZdeg)
    % hold on
    % plot(tv,azmid,'-.')

```

```

% hold off
xlabel('Time, seconds')
ylabel('Azimuth, degrees')
title('Azimuth Angle vs. Time')
subplot(2,2,2)
plot(tv,ELdeg)
% hold on
% plot(tv,el,'-.')
% hold off
xlabel('Time, seconds')
ylabel('Elevation Angle, Degrees')
title('Elevation Angle vs. Time')
% figure
subplot(2,2,3)
plot(tv,aRto)
xlabel('Time, seconds')
ylabel('Radar-Target Range, meters')
title('Radar-Target Range vs. Time')
subplot(2,2,4)
plot(tv,model_no)
xlabel('Time, seconds')
ylabel('C29 Model Number')
title('Data Set Used vs. Time')
figure
plot3(Rtv(:,1),Rtv(:,3),Rtv(:,2))
view([1 1 1]);
grid
xlabel('x-axis, meters')
ylabel('z-axis, meters')
zlabel('y-axis, meters')
title('Path of Target')
hold on
plot3(Ro(1),Ro(3),Ro(2),'*')
hold off
end

```

```

% %%%%%%%%%%% scattered_field.m %%%%%%%%%%%

function [Es,Es_complex,DOA,Pr,fd,AZdeg,ELdeg,azmid,el,model_no,...
vt_old,Rsc_old,pha_old,extent_deg,dvec,d]=...
scattered_field(t,f,Ro,Rt,Rt_nos,vt_old,Rsc_old,pha_old,dt,Dt,...
RPY,Vt,Ptr,Gtr,warnings,scplot,ti,IMvx,IMvy,indiv_mot_on);

% This function calculates the scattered field, Es, in volts/meter,
% and the angular error, in degrees, for the radar
% scattering from a target, as described below. Also calculated are
% the doppler frequency, fd, and the average power received at the
% radar, Pr. The input parameters are the CW frequency f,
% the observation point Ro=[xo,yo,zo], the target location, centered at
% Rt=[xt,yt,zt] and traveling in the direction of the direction vector
% of Dt=[vx,vy,vz]. The target can be traveling with a velocity of Vt
% in the Dt direction. Note that Vt should be a scalar.
% It is assumed that the time dependence,  $e^{(j\omega t)}$  is suppressed for
% the calculation of Es_complex, but is included for Es calculation.
% Thus, the current time, t, should be input in seconds.
% The target is described by the damped exponentials (px,py,a)
% which represent the 2-D spectral domain signature of the target. I
% have assumed that the target is parallel to the xz-plane and pointing
% in the direction described by Pt. Pt is defined as the waterline of
% the aircraft. Pt and Dt may differ if roll, pitch, or yaw are
% present. The power of the transmitter, Ptr, and the antenna
% transmitter gain, Gtr, must be supplied. All of the info concerning
% the radar (or scattering) data the model is based on is given in
% C29model.m, i.e. fmin,df,nf,amin,da,na, where
% fmin=lower frequency of radar data, in Hz
% df=frequency step of radar data, in Hz
% nf=number of frequencies of radar data, integer
% azmid=middle azimuth angle for input data between centerline of
% aircraft and center of radar data, in degrees
% da=angular step, in degrees
% na=number of angular data points of radar data, integer
% el=elevation of data set, in degrees
%
% Author: Joe Sacchini, jsacchin@afit.af.mil, 1 May 96, Australia
% Please see accompanying final report for more detail

% Calculate AZ and EL angle, as seen by the target, from target to

```

```

% observation point
Rot=Rt-Ro; % Observation point to Target distance

% AZ and EL calculations, center of target, traveling in Dt dir, to
% observation point, target is traveling in Dt dir, but can have ROLL,
% PITCH, or YAW, RPY correction
% First AZ calculation assuming no roll, pitch, or yaw
Rotaz=[Rot(1) 0 Rot(3)];
Dtaz=[Dt(1) 0 Dt(3)];
if norm(Rotaz)~=0,
    AZrad=acos(sum((-Rotaz).*Dtaz)/(norm(Rotaz)*norm(Dtaz)));
end
if norm(Rotaz)==0, AZrad=0; end
AZsign=cross(-Rotaz,Dtaz);
AZrad=AZrad*sign(AZsign(2));

% Next, EL calculation assuming no roll, pitch, or yaw
Rotel=[Rot(1) Rot(2) 0];
Dtel=[Dt(1) Dt(2) 0];
if norm(Rotel)~=0,
    ELrad=acos(sum(Rotel.*(-Dtel))/(norm(Rotel)*norm(Dtel)));
end
if norm(Rotel)==0, ELrad=0; end
ELsign=cross(-Rotel,Dtel);
ELrad=ELrad*sign(ELsign(3));

% Now correct AZ and EL for ROLL, PITCH, and YAW
ROLL=RPY(1)*pi/180; PITCH=RPY(2)*pi/180; YAW=RPY(3)*pi/180;
AZradn=AZrad+YAW;
ELradn=ELrad+PITCH;
AZradn=atan2(-sin(ELradn)*sin(ROLL)+sin(AZradn)*cos(ELradn)*cos(ROLL),...
             cos(AZradn)*cos(ELradn));
ELradn=asin(sin(ELradn)*cos(ROLL)+sin(AZradn)*cos(ELradn)*sin(ROLL));

% Convert to degrees
AZdeg=AZradn*180/pi;
ELdeg=ELradn*180/pi;

% Determine physical extent of aircraft for given AZ and EL angles,
% assuming an C29 aircraft fits in a box of dimensions SD=(SX,SY,SZ)
% meters and it is Rot meters away from the observation point

```

```

SD=[15.6 5.37 15.6];
extent=(SD(3)*cos(AZradn)+SD(1)*sin(AZradn)*(abs(AZradn)>pi/4));
extent_deg=(extent/norm(Rot))*(180/pi);

% Determine which C29 model to use
[px,py,a,azmid,el,fmin,df,nf,da,na,model_no]=C29model(AZdeg,ELdeg,f);
%[px,py,a,azmid,el,fmin,df,nf,da,na,model_no]=point_model(AZdeg,ELdeg,f);

% Determine how to use model, what m and n are
no_scatter=length(px);
c=2.9979*10^8;
azmid_rad=azmid*pi/180;
nAZrad=AZradn-azmid_rad;
nAZdeg=nAZrad*180/pi;
darad=da*pi/180;
ko=2*pi*f/c;
Dt=Dt/norm(Dt);
Ruf=c/(2*df);
dfa=f*darad;
Rua=c/(2*dfa);
xstep=Rua/na;
ystep=Ruf/nf;
Dt=Dt/norm(Dt);

ELwarning=5/2;
if warnings==1,
    if abs(el-ELdeg) > ELwarning,
        disp('Warning, Elevation off by')
        disp(el-ELdeg)
    end
end

% Calculate offset from center of grid in 2D frequency space, and
% check for validity of model, m=freq no, n=angle no
m=(f-fmin)/df; % Frequency step
nfmin=-nf/2;
nfmax=nf*1.5;
if m<nfmin, m=nfmin; flagmlow=1; end
if m>nfmax, m=nfmax; flagmhigh=1; end
n=na/2+nAZrad/darad; % Angular step
namin=-na/2;

```

```

namax=na*1.5;
if n<namin, n=namin; flagnlow=1; end
if n>namax, n=namax; flagnhigh=1; end
if warnings==1,
    if flagmlow==1, disp('Warning, frequency too low for valid model'); end
    if flagmhigh==1, disp('Warning, frequency too high for valid model'); end
    if flagnlow==1, disp('Warning, azimuth too low for valid model'); end
    if flagnhigh==1, disp('Warning, azimuth too high for valid model'); end
end
%[m,n]

% Loop through each scattering center
for k=1:no_scatter,

% Randomly displace the scattering center, if desired, independent of
% the other scattering centers (individual motion or non-rigid body motion)
indiv_corr=[0.8 0.2; 0.2 0.8];
if indiv_mot_on==1,
    x_mot=IMvx(ti,k);
    y_mot=IMvy(ti,k);
    corr_mot=indiv_corr*[x_mot;y_mot];
    x_mot=corr_mot(1);
    y_mot=corr_mot(2);
else
    x_mot=0;
    y_mot=0;
end
px(k)=abs(px(k))*exp(j*(angle(px(k))+x_mot));
py(k)=abs(py(k))*exp(j*(angle(py(k))+y_mot));
if scplot==1,
    plot(angle(py(k))*Rua/(2*pi)-0.5*xstep,...
         -angle(px(k))*Ruf/(2*pi)+0.5*ystep, '+')
    hold on
end

% Calculate the range to the k-th scattering center, in 3-D, for DOA
% estimation, azrot is the angle between Dt and the xy-plane while
% elrot is the angle between Dt and the xz-plane. azrot and elrot
% have no dependence on absolute target or observation location, thus
% they are, in general, different than AZ and EL
Rsc1=[-angle(px(k))*Ruf/(2*pi) 0 -angle(py(k))*Rua/(2*pi)];

```

```

    azrot=atan(Dt(3)/Dt(1));
    elrot=atan(-Dt(2)/Dt(1));
% Correct for ROLL, PITCH, and YAW
    ROLL=RPY(1)*pi/180; PITCH=RPY(2)*pi/180; YAW=RPY(3)*pi/180;
    azrotn=azrot+YAW;
    elrotn=elrot+PITCH;
    azrotn=atan2(-sin(elrotn)*sin(ROLL)+sin(azrotn)*cos(elrotn)*cos(ROLL),...
                cos(azrotn)*cos(elrotn));
    elrotn=asin(sin(elrotn)*cos(ROLL)+sin(azrotn)*cos(elrotn)*sin(ROLL));
% now create rotation matrix and determine distance and direction from
% the k-th scattering center to the observation point
    rot_mata=[cos(azrotn) 0 sin(azrotn)
              0 1 0
              -sin(azrotn) 0 cos(azrotn)]; %Rotate in AZ
    rot_mate=[cos(elrotn) -sin(elrotn) 0
              sin(elrotn) cos(elrotn) 0
              0 0 1]; %Rotate in EL
    rot_mat=rot_mata*rot_mate; % Total rotation matrix
    Rsc(k,:)=(rot_mat*Rsc1')'+Rt; %distance from k-th SC to origin
    Rsco(k,:)=Rsc(k,:)-Ro; %distance from k-th SC to obs point
    vt(k,:)=-Rsco(k,:)/norm(Rsco(k,:)); % direction from k-th SC to obs pt

% Calculate damping factor inherent in damped exponential model (may
%be turned off if desired). If this is turned off, then all you have
%is a true point scatterer model. Also, the damping has been limited
%to a factor of 1000 (arbitrary).
    no_damping=0; % to turn off damping, make no_damping=1
    damping=abs(px(k))^m*abs(py(k))^n;
    if damping < 0.001, damping=0.001; end
    if damping > 1000, damping=1000; end
    if no_damping==1, damping=1; end

% Calculate the scattered field due to the k-th scattering center at
%the center of the target
    mag(k)=abs(a(k))*damping;
    pha(k)=angle(a(k))+angle(px(k))*m+angle(py(k))*n;
    Es_s(k)=mag(k)*exp(j*pha(k));

end

% Calculate distance between target center and observation pt, r

```

```

r=norm(Rt-Ro);
ko=2*pi*f/c;
etao=376.7;

% Calculate total scattered field at center of target
Es_complex=sum(Es_s);

% Allow for all power and range calculations
if r~=0,
    Es_complex=Es_complex*sqrt(2*Ptr*Gtr*etao)/(4*pi*r^2)*exp(-j*ko*2*r);
end

% Put in e^{jwt} time dependence and go from spectral to time
Es=abs(Es_complex)*cos(2*pi*f*t+angle(Es_complex));

% Calculate DOA estimate, R here is Ro-Rt, dir from target center
% to obs point
%(working in a target fixed coord-system for DOA calcs)
% NOTE: DOA here means angular error!!!!
d=[0 0 0];
mag=mag*sqrt(2*Ptr*Gtr*etao)/(4*pi*r^2);
R=Ro-Rt;
R_nos=Ro-Rt_nos;
for k=1:no_scatter,
    aa(k)=0;
    for l=1:no_scatter,
        aa(k)=aa(k)+mag(k)*mag(l)...
            *cos((pha(k)-2*ko*sum(vt(k,:).*R))-(pha(l)-2*ko*sum(vt(l,:).*R)));
    end
    d=d+aa(k)*vt(k,:);
end
d=d*(ko/(abs(Es_complex).^2));
DOArad=acos(sum(R_nos.*d)/(norm(R_nos)*norm(d)));
DOA_dir=cross(d,R);
if DOA_dir(3)~=0,
    DOA_sign=sign(DOA_dir(3));
else
    DOA_sign=sign(DOA_dir(2));
end
DOA=DOArad*180/pi*DOA_sign;

```

```

% Calculate estimate of received power, assuming the same receiving
% and transmitting antenna
Pr=abs(Es_complex)^2*(2*pi*Gtr)/(etao*(c/f)^2);

% Calculate the doppler frequency, method one
%dr_dt=(Rt-Rt_delay)/dt;
%vel=(norm(dr_dt,2));
%fd=sum(2*d.*dr_dt)/(2*pi);

% Doppler method two, better method
wd=0;
if ti~=1,
for k=1:no_scatterers,
    d_phi=(pha(k)-pha_old(k))/dt;
    d_dir=sum((vt(k,:)-vt_old(k,:)).*(Rsc(k,:)))/dt;
    d_range=sum((Rsc(k,:)-Rsc_old(k,:)).*vt(k,:))/dt;
    wd=wd+aa(k)*(d_phi-2*ko*(d_range+d_dir));
end
end
dvec=[d_phi d_range d_dir d_phi -ko*(d_range+d_dir)];
if ti==1, dvec=[0 0 0 0 0]; end
wd=wd*(-1/(abs(Es_complex).^2));
fd=wd/(2*pi);
dopp_lim=0; % only use if doppler shift spikey - this is a fudge factor
if dopp_lim==1, % limit doppler spikes due to field drop out
    fdmax=1.5*((2*Vt*f)/(3*10^8));
    fdmin=0.5*((2*Vt*f)/(3*10^8));
    if fd > fdmax, fd=fdmax; end
    if fd < fdmin, fd=fdmin; end
end
vt_old=vt;
Rsc_old=Rsc;
pha_old=pha;

```

```

%          %%%%%%%%%% rigid_motion.m %%%%%%%%%%

function [Rtv_s,Dtv_s,RPYv_s]=rigid_motion(Rtv_nos,Dtv_nos,...
RPYv_nos,tv,dt,trans_on,fc_trans,mag_trans,rot_on,fc_rot,mag_rot_deg);

% Do translational shaking
if trans_on~=1,
    Rtv_s=Rtv_nos; % NO shaking
else % else shake
    nts=length(tv);
    fs=1/dt;
% correlation of x y and z components
    corr_mat=[0.8 0.1 0.1;0.1 0.8 0.1;0.1 0.1 0.8];
% if cut-off freq is under Nyquist
    if fc_trans<(fs/2),
        fc_transn=fc_trans/fs;
        wnx=randn(nts,1)*mag_trans;
        wny=randn(nts,1)*mag_trans;
        wnz=randn(nts,1)*mag_trans;
        b=fir1(16,fc_transn/2);
        wnfxf=filter(b,1,wnx);
        wnfyf=filter(b,1,wny);
        wnfzf=filter(b,1,wnz);
        wnf=[wnfxf wnfyf wnfzf];
    else
% If cut-off freq exceeds Nyquist
        fsn=8*fc_trans;
        up=round(fsn/fs);
        fsn=up*fs;
        wnx=randn(nts*up,1)*mag_trans;
        wny=randn(nts*up,1)*mag_trans;
        wnz=randn(nts*up,1)*mag_trans;
        b=fir1(16,(fc_trans/fsn)/2);
        wnfxf=filter(b,1,wnx);
        wnfyf=filter(b,1,wny);
        wnfzf=filter(b,1,wnz);
        wnf=[wnfxf(1:up:length(wnfxf)) wnfyf(1:up:length(wnfyf))...
            wnfzf(1:up:length(wnfzf))];
    end
    wnf=wnf*corr_mat; % Correlate x y and z movement
    Rtv_s=Rtv_nos+wnf; % Add shake to positions

```

```

end

% Now do ROLL, PITCH, and YAW shaking
if rot_on~=1,
    RPYv_s=RPYv_nos;
else
    mag_rot=mag_rot_deg*(pi/180); % Magnitude of shaking in radians
    nts=length(tv);
    fs=1/dt;
    % correlation of roll pitch and yaw components
    corr_mat=[0.6 0.2 0.2;0.2 0.6 0.2;0.2 0.2 0.6];
    % if cut-off freq is under Nyquist
    if fc_rot<(fs/2),
        fc_rotn=fc_rot/fs;
        wnx=randn(nts,1)*mag_rot;
        wny=randn(nts,1)*mag_rot;
        wnz=randn(nts,1)*mag_rot;
        b=fir1(16,fc_rotn/2);
        wnfx=filter(b,1,wnx);
        wnfy=filter(b,1,wny);
        wnfx=filter(b,1,wnz);
        wnf=[wnfx wnfy wnfx];
    else
        % If cut-off freq exceeds Nyquist
        fsn=8*fc_rot;
        up=round(fsn/fs);
        fsn=up*fs;
        wnx=randn(nts*up,1)*mag_rot;
        wny=randn(nts*up,1)*mag_rot;
        wnz=randn(nts*up,1)*mag_rot;
        b=fir1(16,(fc_rot/fsn)/2);
        wnfx=filter(b,1,wnx);
        wnfy=filter(b,1,wny);
        wnfx=filter(b,1,wnz);
        wnf=[wnfx(1:up:length(wnfx)) wnfy(1:up:length(wnfy))...
            wnfx(1:up:length(wnfx))];
    end
    wnf=wnf*corr_mat; % Correlate ROLL PITCH and YAW movements
    RPYv_s=RPYv_nos+wnf; % Add shake to angles
end

```

```

% Now do direction of flight shaking, if desired
Dtv_s=Dtv_nos; % No change in direction implemented yet

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%          %%%%%%%%% indiv_mot.m %%%%%%%%%

function IMv=indiv_mot(tv,dt,indiv_mot_on,fc_indiv,mag_indiv_mot);

nts=length(tv);
IMv_no=zeros(nts,1);

% Do individual Scattering Center shaking
if indiv_mot_on~=1,
    IMv=IMv_no; % NO shaking
else % else shake
    fs=1/dt;
    % if cut-off freq is under Nyquist
    if fc_indiv<(fs/2),
        fc_indivn=fc_indiv/fs;
        wn=randn(nts,1)*mag_indiv_mot;
        b=fir1(16,fc_indivn/2);
        wnf=filter(b,1,wn);
    else
        % If cut-off freq exceeds Nyquist
        fsn=8*fc_indiv;
        up=round(fsn/fs);
        fsn=up*fs;
        wn=randn(nts*up,1)*mag_indiv_mot;
        b=fir1(16,(fc_indiv/fsn)/2);
        wnf=filter(b,1,wn);
        wnf=wnf(1:up:length(wnf));
    end
    IMv=wnf; % Add shake to positions
end

```

```

% %%%%%%%%% Portion of C29model.m %%%%%%%%%
function [px,py,a,azmid,el,fmin,df,nf,da,na,model_no]=...
        C29model(AZdeg,ELdeg,f,display);

% This function determines which C29-based 2-D damped exponential
% model is best to use for the desired AZ and EL angles. The AZ and EL
% angles are defined relative to the target, the C29. The outputs are
% the target model [px,py,a], the middle azimuth of the data, azmid
% (degrees), the elevation angle of the data, el (degrees), the min
% freq, fmin (Hz), the frequency step, df (Hz), the number of frequency
% samples, the angular step, da (degrees), and the number of angles.
% This function will find the "closest" data set to the specified AZ and
% EL angles. The desired frequency, f, can also be factored into the
% calculation, but is not presently.
% Display images, of original data and scattering center locations,
% and model generated SAR image and scattering center locations, if you
% wish by making display=1.
%
% ALL ANGLES ARE IN DEGREES, NO EXCEPTIONS IN THIS CODE

if nargin<4 display=0; end
if nargin<3 f=(26.0023*10^9)/3; end

% Azimuth and elevation angles of model data
AZmodel=[ -3.76199340820312
-0.08099365234375
3.71899414062500
-3.76100158691406
-0.08099365234375
3.71899414062500
-3.76100158691406
-0.08099365234375
3.71899414062500
20.23899841308594
23.91900634765625
27.71899414062500
20.23899841308594
23.91900634765625
27.71899414062500

```

20.23899841308594
23.91900634765625
27.71899414062500
-20.23899841308594
-23.91900634765625
-27.71899414062500
-20.23899841308594
-23.91900634765625
-27.71899414062500
-20.23899841308594
-23.91900634765625
-27.71899414062500];

ELmodel=[3.00699996948242
3.00699996948242
3.00699996948242
5.00600051879883
5.00600051879883
5.00600051879883
7.00900077819824
7.00900077819824
7.00900077819824
3.00699996948242
3.00699996948242
3.00699996948242
5.00600051879883
5.00600051879883
5.00600051879883
7.00600051879883
7.00600051879883
7.00600051879883
3.00699996948242
3.00699996948242
3.00699996948242
5.00600051879883
5.00600051879883
5.00600051879883
7.00600051879883
7.00600051879883
7.00600051879883];

```

% Determine differences between desired angles and model angles
AZdiff=AZmodel-AZdeg*ones(size(AZmodel));
ELdiff=ELmodel-ELdeg*ones(size(ELmodel));

% Define sorting metric as total angular error
metric=abs(AZdiff)+abs(ELdiff);

% Sort data by metric from actual AZ EL to available model AZ EL.
[sorted_metric,rank]=sort(metric);

% Choose the AZ and EL model with the smallest metric
model_no=rank(1);

% Begin listing model_no 1 through 27, and choose the appropriate model

if model_no==1,
% DATA SET 1
% C29 data, C29_3D_hhN.02700, nkeep=10, dataf=G(1:2:128,1:64)
% AZ= -3.76199340820312
% EL= 3.00699996948242
nkeep=10;
fmin=26.0023*10^9/3;
df=20*10^6/3;
nf=64;
da=0.04;
na=64;
azmid = -3.76199340820312;
el = 3.00699996948242;
energy = 1.0e+02 *[4.46351955889678
3.38992633084404
3.09244961258918
3.03737934157418
2.11876036758480
1.55804105492920
1.44295351235822
1.32096395403193
0.57293375351862
0.50675258067672];
px =[1.00734824451613 - 0.12020931899238i
0.34346787678126 - 1.04892428851494i

```

```

0.19620258114083 - 1.14280994652901i
0.93987525649699 - 0.37095079856928i
0.98559136671387 + 0.20021519900137i
0.83608629828962 - 0.58516977701067i
-0.16587689351521 - 1.02662043344300i
1.00734824451613 - 0.12020931899238i
0.93987525649699 - 0.37095079856928i
-0.12059389273158 + 0.98313482036363i];
py =[0.92461720846197 - 0.38906907824639i
1.00169462678913 + 0.00495474279440i
1.00169462678913 + 0.00495474279440i
1.00169462678913 + 0.00495474279440i
1.00169462678913 + 0.00495474279440i
0.92461720846197 - 0.38906907824639i
1.00169462678913 + 0.00495474279440i
1.00169462678913 + 0.00495474279440i
0.95422339145316 - 0.27814459770801i
0.95773301389047 + 0.32169872488385i];
a =[-0.11870475204384 + 0.13106455941367i
0.00159621059295 + 0.00091296681722i
0.00009043975091 - 0.00002540478449i
0.10690730553061 + 0.14396472753675i
-0.00016535310040 - 0.17792720809369i
0.00076855880197 + 0.08124534399884i
-0.03155655411428 - 0.01035959413232i
-0.07005013611894 + 0.07257767503099i
-0.01078083345972 - 0.09779856718341i
-0.06936266338304 + 0.07446368402615i];
re =[0.10466510944782    0.71533402626833    0.98584320614570];
end

%%%%%%%%% 26 more models are listed in the C29model.m, but not here %%%%

```